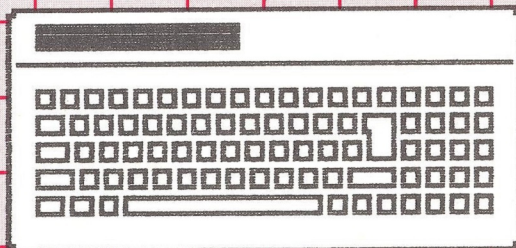
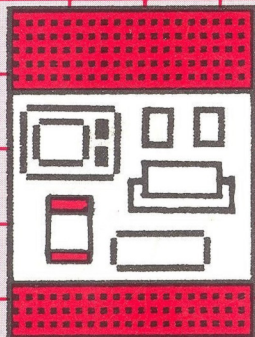
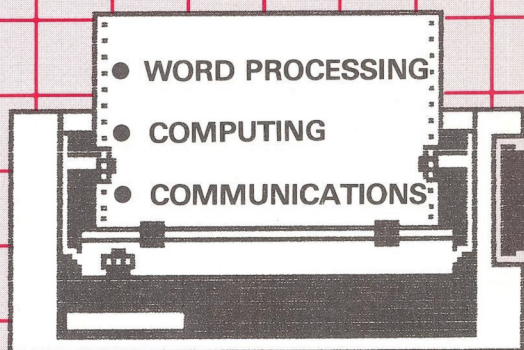
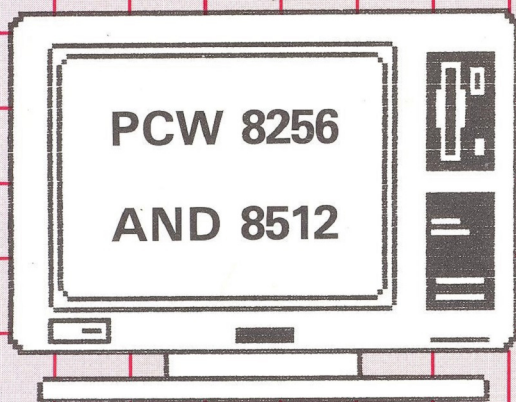


NCC PUBLICATIONS

USING THE AMSTRAD WORD PROCESSOR



MICHAEL MILAN

Using the Amstrad Word Processor

Michael Milan

PUBLISHED BY NCC PUBLICATIONS

British Library Cataloguing in Publication Data

Milan, Michael

Using the Amstrad word processor.

1. Amstrad Microcomputer
2. Word processing

I. Title

652.5'0285416 Z52.5.A4/

ISBN 0-85012-567-7

© THE NATIONAL COMPUTING CENTRE LIMITED, 1986

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, without the prior permission of The National Computing Centre.

First published in 1986 by:

NCC Publications, The National Computing Centre Limited,
Oxford Road, Manchester M1 7ED, England.

Typeset in 11pt Times Roman by Typeface Limited, Portland House,
103 Portland Street, Manchester M1 6DF; and printed by Hobbs the
Printers of Southampton.

ISBN 0-85012-567-7

Introduction

The Amstrad Word Processor is bringing a wide range of computer-based facilities within the reach of non-computer people. It's easy to see why. People in all walks of life need to organise their thoughts and ideas, and to produce them as words on paper. These people can benefit from using a computer and a word processor. Until the Amstrad Word Processor became available, the choice was either a 'professional' computer, or a 'home' computer pressed into service. The former was expensive, beyond the pocket of most people, the latter was not ideally suited to word processing.

The Amstrad Word Processor is *inexpensive*. It is designed *as* a word processor (although it will perform other tasks as well, as this book will explain) and – importantly for many newcomers to information technology – it is complete and ready for use when purchased, with a powerful word processing program and a printer.

Even the recent introduction of the Amstrad Personal Computer has not diminished the attractiveness of the Word Processor for those for whom word processing is the most important requirement. It is still the most inexpensive purpose-built word processor available. The Word Processor has dedicated keys which make many of the features of the software quick and simple to use. The dedicated printer can offer many different type styles.

Having bought this value-for-money word processor, you also possess a very powerful computer which can be used for many different purposes. For example, you can keep databases for your address lists. Graphics programs can be used to produce illustrations and graphs. Spreadsheets can be used for *what-if?* calculations.

A small business could run its entire accounting system on an Amstrad Word Processor. Program packages for these and many other tasks are available, at low cost, for the Amstrad.

As well as all this, the Amstrad Word Processor can be used, with the addition of an adaptor and a modem, as a communications terminal – it can connect to other computers over the telephone. This allows you to send and receive messages and text at electronic speed over long distances, to gain access to computer services, and to obtain information from databases.

The Amstrad Word Processor has been bought by many people who are completely new to electronic computing in any form. They need assistance to do those things for which they originally bought the machine, and also to be made aware of all the other uses to which it can be put.

If you've never used a computer or word processor before, *Using the Amstrad Word Processor* will explain how to get the most out of your machine.

Acknowledgements

I owe thanks to many people who have helped me to put this book together. Several gave me the benefit of their experiences with the Amstrad word processor; others kindly read the manuscript and made valuable comments. I would especially like to mention:

Gabriel Boronet

Brian Hargreaves

Charles Hoskyns

David and Robert Rooke

Bob and Rob Roseveare

Laurie Turner

Edward Yoxen

and my son

Stephen Milan

who wrote the BASIC programs and provided the illustrations.

Thanks are also due to

Digital Research (UK) Ltd, and

Sagesoft plc

for providing review software, and to

Linda Barrett

for her sterling efforts to edit the text and make it more readable.

Michael Milan

February 1986

Amstrad is the trademark of **AMSTRAD Consumer Electronics plc**. LocoScript and Mallard BASIC are the trademarks of **Locomotive Software Ltd**. CP/M, DR Graph and DR Draw are the trademarks of **Digital Research Inc**. Chit-Chat is the trademark of **Sagesoft plc**. Apologies if any other trademarks are mentioned in the book, but not acknowledged here.

Contents

	<i>Page</i>
Introduction	
Acknowledgements	
1 The Information Processor	15
Introduction	15
What is a Word Processor?	16
What is a Computer?	17
Home Computers	19
The Amstrad PCW	20
The Amstrad PCW as a Computer	24
2 Starting Off	27
Introduction	27
Connecting the Cables	28
Loading the Printer Ribbon	29
Does it Work?	29
Loading Paper into the Printer	30
Problems	31
Things to Come	32
3 Preparing Discs	33
Introduction	33
Disc Capacities	34
Buying Discs	35
Protecting Discs	35

	Preparing Discs	36
	Software Back-ups	37
	File Discs	37
	Disckit	38
	Getting into Disckit	38
	Start of Day Disc	43
	Deleting Unwanted Files	45
	Recovering Deleted Files	45
	Disc Name	46
	Group Names	46
4	Creating LocoScript Documents	49
	Introduction	49
	Editing Screen Status Lines	49
	Creating and Editing Documents	51
	Controlling LocoScript	54
	Creating a Document	55
	Standard Templates	56
5	Editing LocoScript Documents	69
	Editing Existing Documents	69
	Codes and Effectors	70
	Phrases and Blocks	72
	Moving Text Around	73
	Saving Phrases and Blocks	74
	Inserting Text	75
	Exit	75
	Back-up Copies	77
6	Printing Out	79
	Introduction	79
	Controlling the Printer	80
	Printing Out Documents	85
	Temporary Stop	86
	How to Abandon Printing	86
	Direct Printing	87
	Using the Printer under CP/M	87

7	Using CP/M	89
	Introduction	89
	Loading CP/M	90
	CP/M Files	90
	CP/M Utilities	91
8	BASIC Programs	99
	Introduction to Mallard BASIC	99
	BASIC Programs	100
	Personal Data	118
9	Commercially Available Software	129
	What Software?	129
	What Will it Cost?	129
	Alternative Word Processors	130
	Database Programs	131
	Spreadsheets	133
	Accounts Packages	138
	Graphics Packages	138
	Communications Programs	140
	Where to Look for Software	142
10	What Are Data Communications?	143
	Introduction	143
	Hardware	144
	Software	145
	Data	145
	Serial Transmission	146
	Standards	147
	Error Checking	148
	Transmission	149
	Buying a Modem	152
	Cables	153
	Using a Direct Connect Modem	154
	Using an Acoustic Coupler	155
	Packet SwitchStream	155

11	Using MAIL232	159
	Introduction	159
	Making a Communications Disc	160
	Terminal Parameters	163
	Making ASCII Files	167
	Transmitting a File	168
	Receiving a File	168
	Using MAIL 232 with Two Disc Drives Fitted	168
	Using an ASCII File in LocoScript	169
	Unformatting a Document	169
	Alternative Communications Software	170
12	Direct Communications	171
	Introduction	171
	Communicating between the Amstrad and a BBC Micro	172
	Communicating between the Amstrad and a Lap Portable Computer	175
	Communicating between the Amstrad and a Portable Electronic Typewriter	179
13	On-Line Services	181
	Bulletin Boards	181
	Mailbox Services	183
	Getting a Mailbox	184
	Other Uses for Electronic Mail	185
	Textnet	186
	Databases	187
	Viewdata	188
	Appendix	
1	Menus	189
2	Character Sets	201
3	Useful Addresses	205
	Index	207

1 The Information Processor

INTRODUCTION

This book is aimed primarily at newcomers to word processing and computing. It will explain, in practical terms, how to start using the Amstrad PCW for three different purposes:

- as a word processor;
- as a computer;
- as a communications terminal (see Chapter 10).

Two models of the Amstrad Personal Computer Word Processor are available. The first one to be released was the PCW8256. This has 256 kilobytes of **random access memory**, and one floppy disc drive. The other model is the 8512. This has 512 kilobytes of random access memory and comes complete with two disc drives. The extra memory may be useful but isn't essential. For instance, it won't allow you to write bigger programs, but you can hold more data in the memory. The second disc drive is quite useful however, and whether you buy an 8512, or upgrade an 8256, this is an option to consider. If the text above doesn't mean anything to you yet, don't worry, it's all explained later in this book.

The Amstrad PCW is a new concept in cheap personal computing. It's not sold to play computer games (although some have been written for it already!). It doesn't *zap* and *pow*. It doesn't have a multi-coloured display. But it *will* run a very powerful word processing program and hundreds of other commercial data processing packages. Although primarily sold as a word processor, it is in fact a powerful computer, and as well as processing words, given suitable

programs (*software*), it will also process all sorts of information. Furthermore, with some additional attachments and the use of the telephone, it can be used as a communications terminal, able to contact other computers and information services anywhere in the world.

WHAT IS A WORD PROCESSOR?

A word processor is essentially a personal computer able to accept, record, arrange and provide text on paper. But pen and paper have been able to do that for centuries, so what has a word processor got to offer that's special? Well, once the pen has written on paper, the message is fixed, and any alteration involves re-writing. Similarly with the typewriter. This allows a more readable presentation of text, but again, once typed, the message is fixed. Correction involves covering over mistakes with opaque fluid, and no rearrangement of the content is possible.

The word processor, like the typewriter, has a keyboard (see Figure 1.1), but pressing the keys doesn't immediately cause a letter to be printed onto paper. Instead, as the keys are pressed the letters are recorded in the word processor's memory and also displayed on a screen. The letters are remembered and can be manipulated. More letters and words can be added, not just after previously stored material but anywhere.

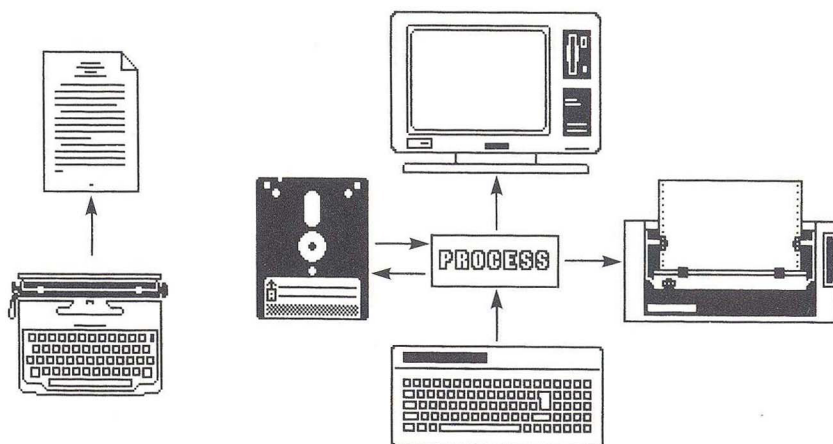


Figure 1.1 Typewriter versus Word Processor

Parts of the text that are incorrect or capable of improvement can be altered until the author is satisfied with the result. Never again will you have to retype a whole page because of an unnoticed mistake. The format can be changed at will. Line lengths, spacings, indentations, etc can be specified, and then altered later until satisfactory. Various layouts can be tried to see which will give the most pleasing result. Then the document can be stored on floppy disc (more about these later), and if required, printed out onto paper.

Hence the most important advantage offered by a word processor is the ability to amend or update stored textual material and to be able to retrieve it at a later date. Only the alterations need be re-typed: what is already stored can be used again. This saves a lot of effort and no extra typing errors will creep into existing text.

If a word processor is really a computer, then to understand it further, we need to look at computers.

WHAT IS A COMPUTER?

Computers are machines for receiving, handling and storing symbols in various ways. The symbols can represent information, words, shapes, sounds, etc. By manipulating the symbols, computers can carry out logical and arithmetical functions which, although individually simple, can be combined in countless ways to enable computers to perform many different tasks.

Computers were invented by human beings, so not surprisingly they have been designed to work in a way rather like that in which we humans work. We receive messages from the world around us through our five senses: sight, hearing, touch, smell and taste. These are our **data inputs**. As we receive these messages we remember them. As we receive more and more messages we can compare them in our memory and make more sense of them. They become **information**. So in this way we can make judgements based on experience. So data is **input**, compared by the intellect with past experiences (stored in the memory), and a decision yields **output**, as shown in Figure 1.2.

This works very well up to a point, but most of us have less than perfect memories. Life in this technological world is now so complicated that few of us can remember every important detail that we

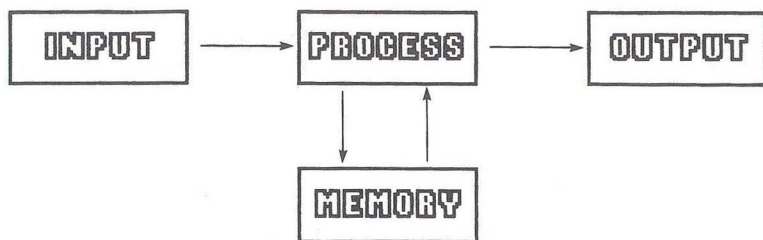


Figure 1.2 Input, Output, Process, Memory

need to know. It's quite easy for us to remember much information in general terms, and limited amounts in fine detail. What we can't do is remember the fine detail in many different areas. Here we often refer to written records. Information that we need to remember accurately, but only occasionally, can be written down for reference when necessary.

Written records range from the note jotted down on a scrap of paper, to large volumes. Libraries are huge repositories of human experience at the disposal of everyone. We may regard this information as stored in files. So files are a limitless expansion for our limited memories, and information processing systems can include files. In this way, files are used to supplement memory (see Figure 1.3).

Files (in the broadest sense) can contain many different types of information. They can contain the instructions to tell us how to do

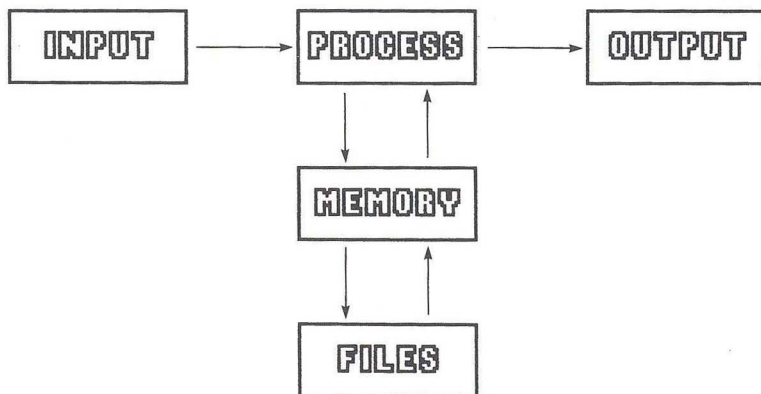


Figure 1.3 Input, Output, Process, Memory, Files

something that we don't know (for example, a cooking recipe). Let's call this a **program** file. Or files can contain facts and figures that we need to know, like railway times or telephone numbers. We'll call this a **data** file. And, of course, a file could contain something to read, a **text** file. Naturally, files are only of use if we can decode the information in them. A car maintenance manual written in Japanese would be of limited use to most Europeans.

But how does all this relate to computers (and indeed, to word processing)? Input, output, memory and files are the common basis for all computer systems, large or small.

HOME COMPUTERS

Everyone will be aware of the tremendous growth of interest in computing in the last few years, and will have noticed the boom in home computers. Until recently a home computer was something designed primarily for playing Space Invaders. Some were used for learning about computer programming in BASIC, a few were used for 'useful' purposes such as word processing or keeping a database. Some of the word-processing programs written to run on home computers are very good, but they're restricted by the *machines* on which they run. Limited memory is usually a problem and perhaps the screen layout doesn't allow the text to be seen in what will be its printed form. Many home computers rely on some form of cassette-tape storage which is far from ideal. Finally, at the home-micro end of the market, printers usually have output that is less than 'correspondence quality'.

Normally a home computer is bought piece by piece to make up an outfit that suits the needs (and pocket) of the owner. First there is the computer itself, which usually includes the keyboard. The screen may be a domestic television set pressed into service (poor definition), or else a special monitor (more expensive). The data storage may be an ordinary cassette recorder (impractical for serious use), or a disc drive (often as expensive as the computer itself). After that, there is still a printer to buy, another very costly item if the print quality is to be good enough for anything other than listing out programs. All these items have their own mains leads, and are joined together by a collection of cables which could put a telephone exchange to shame!

THE AMSTRAD PCW

The Amstrad PCW, on the other hand, comes in one box. This contains all the essential elements necessary for serious computing applications. It has ample memory (256 or 512 kilobytes depending on the version you buy), a dedicated green-screen monitor, a disc drive and an excellent printer. There are just three units, all driven by just one mains lead, and with three short interconnecting cables. And all this at a price which we are used to considering reasonable for just the computer, not all the extras as well. Once out of the box, add a mains plug, and it's ready to go. With a look at the bulky manuals supplied, a little practice, *and* some help from this book, you will be able to word process so easily that you'll never want to touch an ordinary typewriter again.

Input

Just as humans have several ways of receiving input (sight, hearing, smell, touch, taste), so there are several ways that information can be fed into your Amstrad, as shown in Figure 1.4.

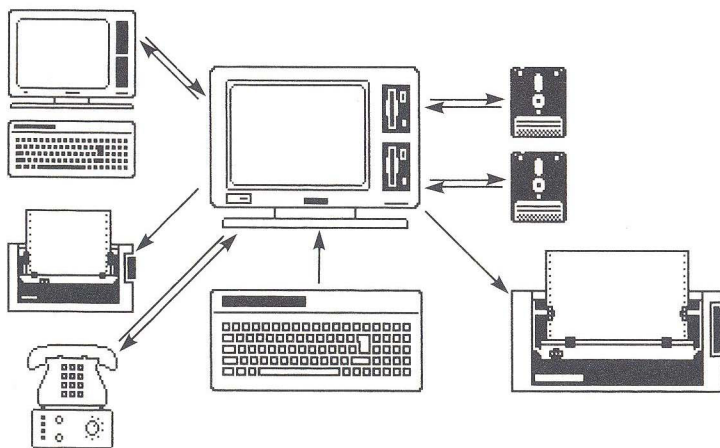


Figure 1.4 Inputs and Outputs

In word processing, the most common **input** route is through the keyboard. On the Amstrad there are two other ways, from the disc

drive(s) and through the optional RS232 adaptor. This will allow information to be received from other computers, perhaps over the telephone in the form of electronic mail (see Chapter 10).

Output

There are four methods of **output** used in the Amstrad word processor. They are – to **screen**, to **disc**, to the **printer**, and via the optional **RS232 adaptor**. As text is typed, the immediate form of output is via the screen. In this way you can see what has been entered, decide on layout, see if any corrections are needed, etc. When the text is finished, it is then output to a disc file for future editing and printing. You can also choose to print it at the same time, or else print later directly from the disc file. Through the optional RS232 adaptor it's also possible to output the file to a different printer or a different computer (although not directly from LocoScript). This can be done directly, or through a modem over the telephone network to a distant computer.

So how does a computer memory compare to a human memory? First of all it's essential to remember that a computer is just a machine that can obey instructions. All computer systems consist of two essential elements, **hardware** and **software**. The *hardware* is the machine (or one of its add-ons). *Software* is the instructions that someone has written to tell the hardware what it is required to do. The hardware has no inherent intelligence of its own, the software has to tell it in the minutest detail what to do under all the circumstances that it is required to operate. Even the most complex computer is a combination of hardware and software.

Memory

The memory of the Amstrad stores information, although this has to be in a form that a microcomputer can handle. At heart, computers are very simple. The memory has a very large number of memory locations, each one of which can store a minute electrical charge. Each location can be either charged or not charged, and this condition can be changed at will. These basic units of information are known as **bits**. The state of a bit is usually represented as a 1 or a 0. These may not seem a very promising basis for an information system, but if bits are grouped together, then their combined values can start to have greater meaning.

The Amstrad is one of the large family known as **eight-bit computers**, which means that the bits are grouped together in eights as the *normal* unit of information. Each group of eight bits is known as a **byte**. Now, using binary arithmetic, the bit pattern of each byte can represent a value between 0 and 255. These are the 'characters' that are manipulated in the computer, whether in word processing, CP/M, BASIC programs or whatever. Depending on its value, a byte might be interpreted as a letter of the alphabet, a single number, or perhaps some control code to trigger an event in the computer. Appendix 2 shows how some of the values are interpreted by the Amstrad in various modes.

Memory is usually measured in bytes. One byte is the amount of memory space needed to record one letter, character or digit. A kilobyte is 1024 bytes (2^{10}). To give you an idea of the scale of memory, Figure 1.5 shows the amount of text that can be stored in one kilobyte.

A Aback Abaft Abandon Abandoned Abandonment Abase Abasement
 Abash Abate Abatement Abess Abbey Abbot Abbreviate Abdicate
 Abdication Abdomen Abduction Aberration Abet Abettor Abeyance
 Abhor Abhorrence Abide Ability Abject Abjure Ablaze Able
 Ablution Aboard Abode Abolish Abolition Abominable Abomination
 Aborigines Abortion Abound About Above Abreast Abridge
 Abridgement Abroad Abrupt Abscess Abscond Absence Absent
 Absentee Absolute Absolutely Absolution Absolve Absorb
 Absorption Abstain Abstemious Abstemiousness Abstinence
 Abstract Abstracted Abstraction Abstruse Absurd Absurdity
 Abundance Abundant Abuse Abusive Abyss Academic Academician
 Academy Accede Accelerate Acceleration Accent Accentuate
 Accept Acceptable Acceptance Acceptation Access Accessible
 Accession Accessory Accident Accidental Acclamation Accomodate
 Accomodating Accomodation Accompaniment Accompanist Accompany
 Accomplice Accomplish Accomplishment Accord Accordingly Account
 Account Accountable Accredited Accumulate Accumulation
 Accuracy Accurate Accusation Accuse Accuser

Figure 1.5 Approximately One Kilobyte of Text

Inside the computer the memory is in two forms. There is some permanent memory so that when it is switched on it will know what to do. The Amstrad PCW has a minute amount, 256 *bytes*, one thousandth of its total memory capacity, just enough to tell it to try to load instructions from the disc drive. The rest of the information, the

program and the data (eg the document being word processed), is held within the computer in **RAM** (Random Access Memory). Actually, random access memory isn't a very good name to describe this sort of memory, because other types of computer memories can be randomly accessed too. A better name would be Read/Write memory. In RAM there is a minute framework to hold the information as very small electrical charges. RAM has the advantage that it is quick and easy to write information to it, read and if necessary, to re-write. But the snag is that if the power to the machine is switched off, however briefly, all the information is instantly lost.

Normally, eight-bit micros can only have 64 kilobytes of memory, that is they can only store 65,536 characters at one time. This is because 65,536 is the largest number of memory addresses that an eight-bit processor can point to. However the PCW8256 has a massive 256 kilobytes of RAM (and there's even more – 512 kilobytes – in the PCW8512), but gets over the problem by splitting the memory into different blocks and addressing each of these separately.

Although there are 256 or 512 kilobytes of RAM, not all of this is available to the user for programs or files. The operating system and the language will take up quite a lot of memory space. In the case of *LocoScript* this is about 82 kilobytes. CP/M and BASIC will need 68 kilobytes. Then another 102 kilobytes (356 kilobytes in the PCW8512) is set aside as a *memory disc*. This is an area of memory that appears to the operating system as another, very fast, disc drive. Finally there is the working space, known as the *transient program area* (TPA) which the user can use for a document or program and variables.

So, inside the Amstrad we have a small amount of permanent memory which cannot be changed, or temporary memory which allows us to enter different programs for different purposes (such as the word processing software) and to create different documents. The problem with this temporary memory is that the information disappears if the power is switched off. Now if all the information that computers and word processors need had to be entered at the keyboard each and every time that it was needed, then it's quite likely that they would never have caught on! To get over this problem we use a means of storing information in files, so that it can be loaded into the memory when it's needed.

Files

The Amstrad word processor uses three-inch micro-floppy discs to hold all the files of word processed documents, data or programs. Information from the Amstrad's memory can be copied directly onto a disc. Provided that nothing untoward happens to it, the information is now stored until required again, when it can be quickly copied back into the Amstrad's internal memory. As the discs are relatively inexpensive, and can each hold a lot of text (especially discs for the second drive, which is optional on the 8256 but standard on the 8512), the storage capacity becomes virtually infinite. If information on a disc becomes out of date or unwanted it can be overwritten with new information. Discs don't fail very often. Unfortunately when they do it's without warning. The results can be catastrophic, so always be sure to keep a back-up copy of any information that you will want again.

When creating or editing a file, LocoScript requires enough free space left on the disc to store it. So when editing a file you need to be sure that there is enough free space to store the old file, plus any new additions. The file sizes and the free disc space can all be seen on the disc management menu.

Each disc for the upper (or only, on the 8256) disc drive has a capacity of 170 kilobytes per surface. The discs are reversible, so each disc has a total capacity of 340 kilobytes. The second disc drive of the 8512 (which may be purchased for the 8256) fits below the first one. Discs for this drive hold 706 kilobytes over both sides, which are treated as one surface. If you count all the available capacity of both drives, and the memory disc it's possible to have almost *one million* characters available in the computer at one time. Figure 1.6 illustrates various internal memory input and output options.

THE AMSTRAD PCW AS A COMPUTER

The Amstrad PCW is not just a word processor, it's also a powerful personal computer. At the heart of any personal computer is the *microprocessor*, the chip which makes all the calculations to make the computer work. The microprocessor used in the Amstrad is the *Zilog Z80*. This allows it to run under the industry-standard operating system *CP/M*, for which literally thousands of software programs have been written. LocoScript, the word processing soft-

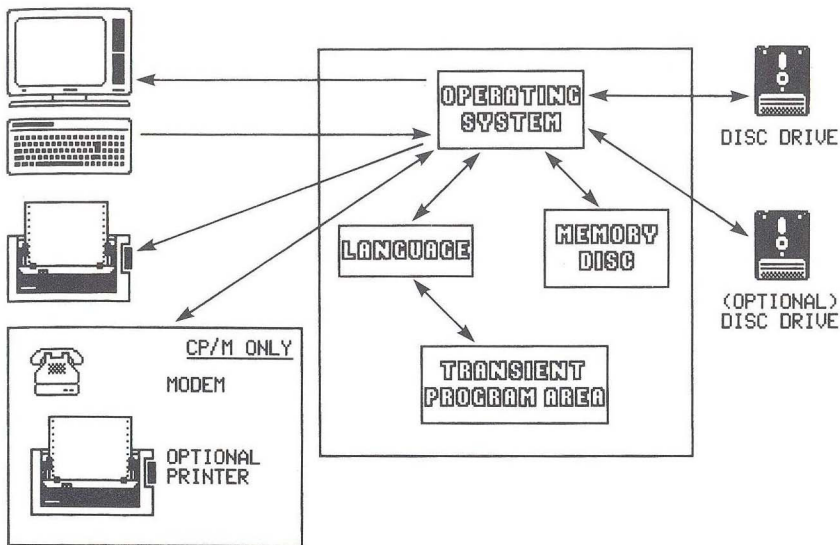


Figure 1.6 Amstrad PCW - Internal Memory, Inputs and Outputs

ware supplied as standard, does *not* run under CP/M. This is a slight disadvantage because it means that to go from LocoScript to CP/M or vice-versa means completely changing the contents of the Amstrad's memory, thus losing any information that may be stored there.

CP/M also comes with the basic kit. The version supplied is CP/M PLUS (also known as CP/M 3). Also supplied are many CP/M utility programs (many of which will be explained in Chapter 6), and two programming languages, *BASIC* and *DR LOGO*. BASIC and LOGO programming are beyond the scope of this book; however there are several useful BASIC programs listed in Chapter 8.

Software

Unlike nearly all other personal computers of recent times, the PCW contains almost no software permanently. There is practically no *read only memory* (ROM). It has a very large *random access memory* (RAM) (by 8-bit computer standards), and all software is loaded in off disc. This makes the machine very versatile, and also

makes updating the software a very easy and inexpensive process. This is just as well, because as with any new system, early versions of the LocoScript software were not without their bugs! The first machines came with LocoScript version 1.0. To see what version you have, you will need to look quickly as the LocoScript program is loaded. The version number is shown on the first title screen while the second part of the program is loaded. This book was written with version 1.2 which is better than the previous ones. If you have an early version you should see your dealer about an upgrade.

Many software suppliers are adapting existing CP/M programs for the Amstrad, and some new software is being written. With so much software readily available, the Amstrad is capable of much more than word processing. It can process information of all sorts.

2 Starting Off

INTRODUCTION

Here is a list of the things that you must do after getting your new Amstrad word processor, before you can start using it properly. Although you are probably itching to try it out, it really will save you time and frustration to do the necessary things first. It will also help you to learn something about your new machine before you start using LocoScript.

Things to do straight away.

1. Take out of box and check all the bits.
2. Fit mains plug with 3 A fuse.
3. Fit printer ribbon.
4. Switch on and test.
5. Format some discs.
6. Make back-up copies of the software discs.

Things to do soon.

7. Prepare start of day disc.
8. Prepare LocoScript working discs.

The whole kit is supplied in one large carton. Open the box carefully and after taking out the various parts (see Figure 2.1), examine the packing to make sure that no small items are left behind. There is

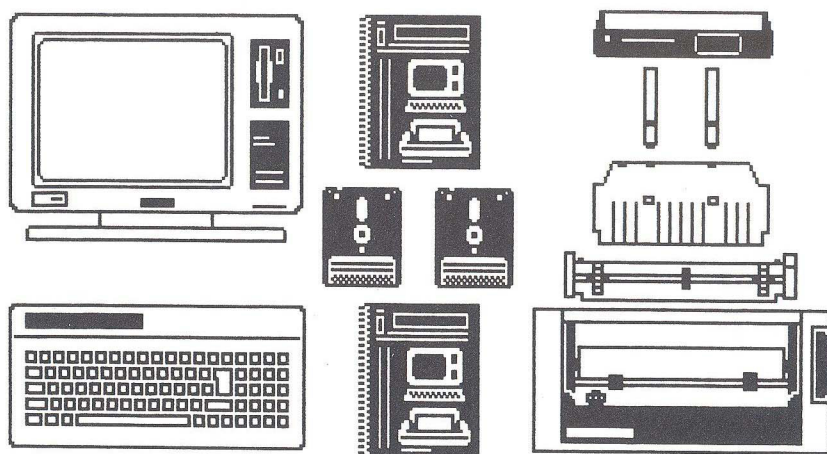


Figure 2.1 What Do You Get?

only one mains electricity lead: if you are using the standard 13 amp UK mains plug make sure that it's fitted with a 3 amp fuse.

CONNECTING THE CABLES

Apart from the mains lead, there are three other cables to be connected. The keyboard is connected to the monitor unit through a lead of generous length with a DIN plug on the end. The plug has just four pins, and is plugged into the monitor unit at the bottom of the right hand side. The plug will only fit in one way, make sure that it is the right way round and don't force it.

If the keyboard cable is generously long, the printer cables are only just long enough. There are two, a power cable (carrying only 24 volts) with a small concentric plug on the end, and a flat ribbon cable to carry the data. They both plug into the top of the back of the monitor unit, on the same end as the keyboard socket. The printer has to be placed to the right of the monitor unit, the leads aren't long enough for it to go anywhere else. Very early models of the Amstrad were even worse, as the cables were connected to the printer in the middle of the back which meant that they tended to get in the way of continuous paper. If the short length of the printer cable gives you a real problem, then RSD Connections Ltd, PO Box 1, Ware, Herts make a 1-metre extension.

LOADING THE PRINTER RIBBON

The ribbon needs to be fitted into the printer before any printing out is done (see Figure 2.2).

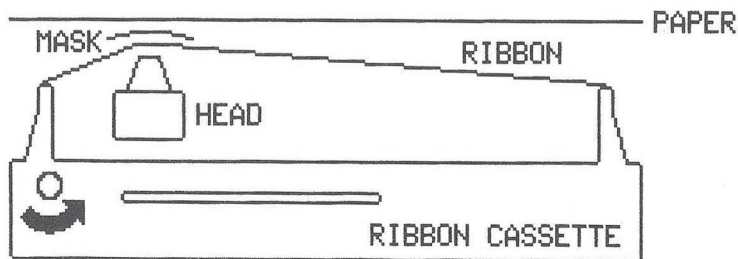


Figure 2.2 Fitting the Printer Ribbon

The printer is made by the Japanese company Seikosha. If you can't get a ribbon specifically marked for the Amstrad, the equivalent is for a Seikosha SP-80051. The ribbon cartridge is very similar to, but unfortunately (annoyingly) slightly different from, the standard Epson type, so be sure when you buy a replacement one that it fits the Amstrad printer. To fit the cartridge, first turn the small knob on it anti-clockwise to take up any slack in the ribbon. Then place the cartridge in the printer in front of the print head so that the ribbon passes between the print head and the mask. If it's correctly positioned, the cartridge should click into place. Turning the knob anti-clockwise again will allow the ribbon to straighten out and pass correctly over the print head.

DOES IT WORK?

At last the moment when you can switch on and see your new word processor working. Check that all the cables are connected properly, plug in and switch on at the switch marked **Power** at the bottom left hand corner of the screen. If all is well, the first thing that you will see is the small red light at the bottom right of the disc drive glowing dimly. Then gradually, as it warms up, the video display screen will become light green. If not, first check that the power is switched on at the wall!

Look at the software discs supplied and take the one marked:

LOCOSCRIPT

Place it in the upper disc drive, the end opposite the label first, with the LocoScript side towards the screen. It should go right in. Within a couple of seconds of it clicking into place the red light should glow more brightly, the disc drive should whirr and a series of lines should appear starting at the top of the screen. Within another seven seconds the screen should clear and the LocoScript title appear. More whirring (including the printer this time) for about 16 seconds, the screen will change again and you'll be in the *disc management menu*. But that's quite enough for now, before you go any further you should read Chapter Three and make back-up copies of your precious software discs. Don't try to use the printer without loading some paper into it first.

LOADING PAPER INTO THE PRINTER

There are two ways of feeding paper into the printer. If you want to use single sheets of paper, then there is a paper tray which slots onto the printer behind the roller. This helps to guide single sheets in, and also holds them upright during printing. There are two small extension bars which make it taller. The paper tray will hinge forward when the printer is not in use to act as a dust cover. The printer has an excellent 'auto-feed' feature when loading paper. When the paper loading knob (see Figure 2.3) beside the paper feed knob is turned one quarter of a turn towards the user, the print head moves to the centre of the printer and the paper is automatically fed to the start position. This only works when software is loaded into the Amstrad. The paper feed works very well except that it isn't strong enough to handle envelopes. The remedy is to use adhesive labels (which can be bought on sheets to fit in the printer), or else window envelopes (which save typing the envelope out separately).

If you want to use continuous computer-type stationery, then the paper feed tray has to be removed and the tractor feed attachment fitted in its place (see Figure 2.3). This is clicked into position on the top of the printer, two hooks at each end of the front of the tractor being fitted into holes, the back is then swung down and clicked into place. The two tractor wheels slide from side to side to accommodate any size of paper up to the standard 9½ inches (24 cm) wide. The paper loading knob is used in the same way as with single sheets to

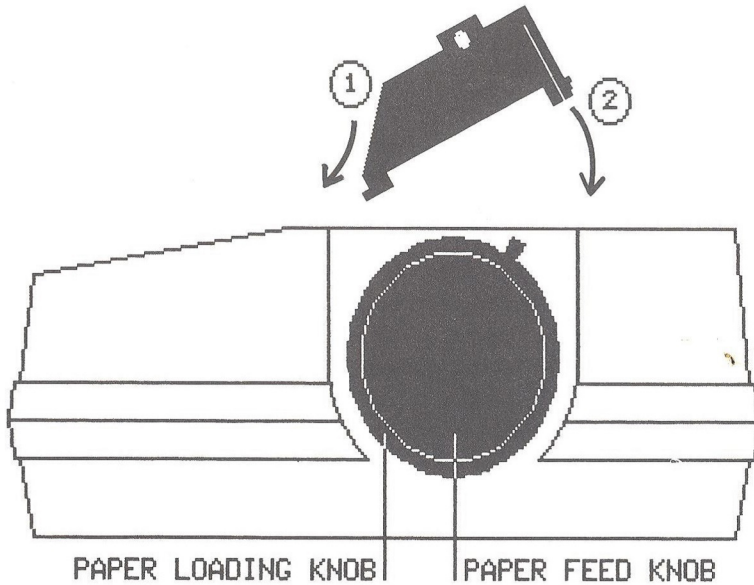


Figure 2.3 Paper Loading Knob and Fitting of Tractor on Printer

feed the paper around the roller; when the paper feed knob is pushed back again, the paper is freed to be positioned onto the tractor wheels by hand. When using continuous stationery, the paper feed tray can be fitted *back to front* (that is with the ribs to the back of the printer). This helps to keep the paper coming out of the printer away from that entering it.

PROBLEMS

One of my friends bought an Amstrad word processor which promptly blew up in a cloud of smoke when he switched it on! Fortunately this is a *very* rare thing to happen and let's hope it won't happen to you. But it does illustrate the usefulness of having *yours* demonstrated in the shop *before* you take it home. A much more likely occurrence is that *nothing* happens. If this is the case, then a systematic approach is best, because there's probably a simple explanation which will save a trip back to the shop.

The first thing to suspect is the power supply. Is the socket

switched on? Is the fuse blown? Some caution is needed here: it may have blown for a good reason. Check the wiring of the plug. Is there power at the socket? Try another socket, or test the socket with another appliance that's known to be working.

If the screen glows but LocoScript fails to load, it may be because you waited too long before you put the disc in. The Amstrad looks for a disc for about 100 seconds, after which it gives up and becomes dormant. There's no need to switch off and on again to wake it up. Just press the <SPACE> bar. If that isn't the problem, try the other side of the disc which has CP/M. This should automatically load in the same way as LocoScript.

The absolute minimum size of worksurface for the machine is about 36 inches (90 cm) by 20 inches (50 cm) deep. At this size, the screen may be uncomfortably close, so a desk or table of this size should be away from the wall, so that the monitor unit can hang out over the back. Also, if you choose to use continuous computer-type stationery, this will need to be kept underneath the table, and fed up the back into the printer, or the worksurface will need to be correspondingly deeper. If you need to make some space in front of the Amstrad (to write for instance, you do occasionally need to do this, even with a word processor), the keyboard will sit very neatly out of the way on top of the main unit.

THINGS TO COME

The Amstrad word processor is sure to be a popular machine, and you can be sure that a lot of firms will offer hardware enhancements for it. Already from Amstrad there is the second disc drive (which comes as standard with the PCW8512) and the Centronics/RS232 serial port. What about a mouse? Watch the computer magazines for developments.

Chapter Nine tells you about the sort of software that you can get, but even LocoScript is capable of improvement. Now that it works well (version 1.2) the major item for attention is speeding up the way that the cursor scrolls through the text. It may also be that a new version is introduced which incorporates **mail merge** and a **spelling checker**, and the ability to use an alternative printer.

3 Preparing Discs

INTRODUCTION

The first floppy discs were introduced by the computer giant IBM as a means of entering information into mainframe computers. They were known as *floppy* discs because the discs were flexible, and enclosed in a thin card case. Those discs were 8 inches in diameter. They were later used for many microcomputers, but overtaken in popularity by *mini* floppy discs which were similar, but 5¼ inches across. The disc drive built-in as standard on the 8256 (the upper drive on the 8512) is of the type designed by the Japanese company Hitachi. The discs are known as 3-inch **compact-floppies**, and the format is known as **CF2**.

The floppy discs themselves are only 3 inches (actually 72mm) in diameter, but they are enclosed in an oblong rigid plastic case, which is why they don't look like discs and don't appear to flop (see Figure 3.1). The dimensions of the case are 100mm long by 80mm wide, and 5mm thick. Outside the disc drive the actual disc itself is pro-

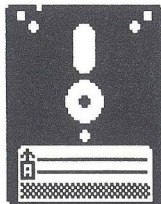


Figure 3.1 Compact Floppy Disc

tected by a metal shutter. Once inside the computer, this is drawn aside to allow a head to touch the disc and read or write the magnetic signals. The rigid case and shutter make the compact floppies much more robust than the earlier types, and they can be quite safely carried in the pocket or posted.

DISC CAPACITIES

Despite their relatively small size, these discs can hold a great deal of information (see Figure 3.2). One compact-floppy disc in the standard (upper) disc drive of the Amstrad can hold 344 kilobytes of information, which is the equivalent of about 150 pages of this book. The (optional) second disc drive uses a different recording technique allowing twice as much information to be recorded on each surface of the disc. It also has two read/write heads allowing both sides of the disc to be treated as one surface. A disc in this drive can hold far more text than this book, the equivalent of about 300 pages.

Within the plastic case, the actual disc is made of material similar to, but stiffer than, the tape in an audio cassette. Unlike the audio cassette however, most discs have a magnetic coating on both sides. Although the standard disc drive in the word processor can only

	Upper drive	(Optional) lower drive	Memory disc
Number of tracks	80 (40 per side)	160 (80 per side)	—
Sectors per track	10	10	—
Bytes per sector	512	512	—
Usable bytes per surface	172k	706 (treated as one surface)	102k/356k*
Usable bytes per disc	344k	706k	—
Files per surface	64	256 (treated as one surface)	64

* PCW8512

Figure 3.2 Disc Capacities

read one side of the disc, the discs can be turned over, so that both sides can be used to store files.

BUYING DISCS

3-inch compact-floppies as used in the Amstrad are not the only small size of floppy disc available. Several sub-5¼-inch formats were designed, but only two have survived onto the marketplace. These are the 3½-inch format designed by Sony and the 3-inch type designed by Hitachi. The Sony disc has been used by many more computer manufacturers than the Hitachi one, and the format is known as *micro-floppy disc*. As a result, many more floppy disc manufacturers make discs in that format. However, the popularity of Amstrad computers will make sure that 3-inch compact-floppy discs stay in demand. But do make sure that you buy the right size.

There are two sorts of compact disc, CF2 and CF2-D. The standard disc drive in the Amstrad takes the CF2 type: these are double sided, but designed to be turned over in disc drives with only one read/write head. The (optional) second drive needs the CF2-D type which are designed for drives with two heads, where the disc is only ever rotated in one direction. The manufacturers advise against turning over CF2-D discs in single headed drives. Even so they seem to work alright, but may not be reliable in the long run. CF2-D discs are sometimes referred to as CF2DD discs; that is CF2 *double density*.

It's unfortunate that discs from the two drives are not completely compatible. However the vast capacity of drive B is very useful. This drive is able to **read** discs written in drive A, and this is useful as it makes copying CF2 discs very quick and straightforward. However drive B can't **write** to discs formatted for drive A, and drive A can't read or write to discs formatted for drive B. As the physical dimensions of the discs for either drive are identical, you'll need to carefully mark each disc with the drive for which it's formatted.

PROTECTING DISCS

Once the disc is in the drive, the metal shutter protecting the disc is slid back so that the read/write head can come into contact with it through the head-slot. The shutter also uncovers the small hole opposite the read/write head which, as the disc rotates, allows a

beam of light and a photo-cell to detect the rotation of the disc. Near to the centre of the disc is a small hole which allows the beam of light to fall momentarily on a photo-cell once per revolution.

At the end of the disc case opposite to the label are four more holes. The two larger ones are used to locate the disc accurately in the drive. The others, the small ones furthest from the label, are the **write-protect** holes. They act rather like the record prevention tabs on audio and video cassettes. Each of these protects one of the surfaces of the disc from accidental erasure. If the appropriate hole is closed, it allows the disc to have new information recorded onto it. The holes are closed with small shutters which are incorporated into the disc case. These usually need something like the tip of a pencil to open and close them. When the holes are opened, the programs and files recorded on the disc are protected from erasure. The shutters can be closed again if you no longer want to protect the recorded material, although some discs on which commercial software is provided have the shutters missing. If you hold the disc case with the label at the bottom and the two write-protect holes at the top, the left hand hole protects the disc surface facing you. They usually have an arrow embossed into the plastic disc casing (see Figure 3.3).

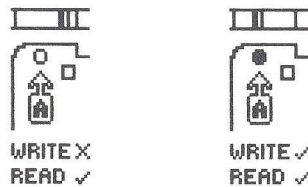


Figure 3.3 Write-Protect Holes

PREPARING DISCS

Before you use your word processor it will be necessary to prepare some new floppy discs for use. There are two sorts that you will need, **back-up** copies of the program discs supplied with the word processor, and **file discs** – empty discs to hold the documents that you are going to create.

SOFTWARE BACK-UPS

Even though your word processor uses 3-inch compact-floppy discs which are more robust than the older, larger types, you must not expect them to be totally secure. Even though there is less likelihood of something going wrong, it's still possible for them to fail. So it's essential to keep **back-up copies** of any discs that have software programs and documents.

The discs that are likely to get the most use are those that hold the software supplied with the outfit. Because the Amstrad word processor has no internal programs, these must be loaded from disc every time that the system is switched on, or when you want to change from the word processor program to CP/M and vice versa. So it's very important to make a back-up copy of the software supplied as soon as possible. Don't attempt to use the supplied LocoScript disc to record your own documents. In fact it's protected so that you can't record on it. Use the back-up copy for everyday use, if that fails, make another back-up from the original. Usually, if a disc fails, it's only the magnetic record on the disc that has become faulty (corrupted), quite often another copy of the original software can be made onto the same disc. Certainly it's worth a try before you throw away an expensive disc.

When you create back-up files of the software, it may not be necessary to copy *all* the files on the disc. The software discs supplied with the Amstrad word processor hold many files that won't be needed very often. The back-up discs that you make for everyday use are better without these, as we shall see in *Start of day discs* later in this chapter.

FILE DISCS

You will also need to create some discs on which you can store the documents that you will be typing. A new disc, when it's first taken out of its packet, is completely blank. It's like a car park with no spaces marked. It has no storage structure. If we are to store programs and files on the disc, then we need to *format* it so that the word processor knows where to store the information. No less importantly, it must be able to find that information again later when you want to re-load it back into the computer. It's useful to think of a disc as the drawer of a filing cabinet. This would have spaces in which to

store documents, and these can be marked for easy later retrieval. These spaces are called files. So we call anything stored on the discs **files**. These could be documents or computer programs.

Formatting a disc simply fills it with magnetic reference marks so that the disc drive can store recorded information accurately. Making a back-up copy of a disc automatically formats the new disc.

DISKIT

There is a utility program provided with the word processor called **Diskit** which will help you make the back-up copies that you need, and format new (and corrupted) discs.

Diskit won't work while LocoScript is loaded in the Amstrad, it only works under CP/M. So if you are going to create a document, make sure that you have enough space for it on a suitable formatted disc ready in advance. If you have to leave LocoScript in order to go into CP/M to format a new disc, you will lose any information in the machine's memory.

GETTING INTO DISKIT

Make sure that the Amstrad is switched on. Select the disc supplied marked **CP/M PLUS** (this is on the other side of the disc with LocoScript). There will be more about using CP/M in Chapter 7. Here is just enough to help you use Diskit. Place the chosen disc in the disc drive (the top drive if you have two) with the CP/M side to the left, facing the screen. If the computer has no program running already, CP/M should load automatically. If the computer was already on, with perhaps LocoScript running, then the CP/M won't be loaded until you press the <SHIFT>, <EXTRA> and <EXIT> keys (hold down the <SHIFT> and <EXTRA> keys and then press the <EXIT> key before you let them go). This causes CP/M to *boot up* – that is, load itself into the computer's memory and then run.

If by chance you have put a disc into the drive without any systems files (neither CP/M nor LocoScript), then on trying to load it, the computer will make a *beeping* noise. When you put in the correct disc, pressing <SHIFT>, <EXTRA> and <EXIT> will have no effect. In this case, simply press the <SPACE> bar and the computer

will load automatically. As mentioned in the last chapter, a similar thing happens if you switch the Amstrad on, and don't put a disc in straight away. The machine looks for a disc every second or so, but after about a hundred seconds it gives up. If you put a disc in after that, you need to press the <SPACE> bar to wake it up.

The computer's screen will go light green, and then horizontal lines will appear downwards from the top. The screen should then accounce:

```
CP/M Plus Amstrad Consumer Electronics plc
v 1.4, 61K TPA, 1 disc drive, 112K drive M:
A>
```

This tells you that you are in the CP/M Plus operating system. If the screen doesn't show this, check that you have inserted the correct disc in the drive, and if so, that it's the correct way round. If you have two disc drives fitted, or the Centronics/Serial adaptor then the message will show this.

A> tells you that the computer regards drive A, the standard (upper) disc drive as the **default** drive. *Default* is a word used for settings which will apply if no instructions to the contrary have been given. So drive A is the one where the computer will look for files, unless instructed to do otherwise. Now type:

```
diskit<RETURN>
```

This tells the computer to load and run a program called **diskit**. The command is simple because the diskit program is on the same disc as CP/M, and so is already in the *default* drive. After the usual whirring noise from the disc drive the screen will clear and four boxes will appear, representing the function keys <f1>, <f3> and <f5> and the <EXIT> key. The three function keys each perform a different function when pressed:

```
<f1>    Verify
<f3>    Format
<f5>    Copy
<EXIT>  Exit from the program
```

<f3> Formatting Discs

Formatting a disc is quite straightforward, but remember that for-

matting will erase any files already held on the disc. The quickest way to clear an already used disc is to re-format it, but be sure that any disc that you format **doesn't** have on it any files that you want to keep.

Place the disc in the disc drive with the surface that you want to format towards the screen. Press <f3>. You will be asked to confirm that you **really** want to format that side (you can't say that you weren't asked!). You do this by pressing the <Y> key. Pressing any other key will abort the formatting. Once confirmed, the screen will clear and a number in the top left hand corner will show progress as the computer formats the 40 or 160 tracks on the disc.

If you have two disc drives fitted, then the program will give you the option of formatting a disc in either drive.

Formatting with Two Disc Drives

With two disc drives fitted, when you press <f3> for copy, you are faced with another menu page which gives you the choice of formatting a CF2 disc in drive A:, or a CF2DD disc in drive B:. Remember that CF2DD discs only go in the drive one way round (although there is nothing to stop you putting them in the wrong way). It's best to adopt the convention of always formatting them with side A or 1 towards the screen. Both surfaces are formatted at one go, and if you listen to the drive as formatting takes place you will realise that the head only moves for even numbered tracks. Therefore the drive is continually switching from one head to the other.

<f5> Copying Discs

Disckit can copy the entire contents of one disc surface onto another. The disc that you want to copy is known as the **source disc**, and the disc onto which you want the information copied is known as the **destination disc**. The destination disc can either be completely blank, or else already formatted, or even a disc that has already been used to store information. As with *format*, any information already on the destination disc will be erased. So check carefully on the disc directory (see Chapter 4 for LocoScript discs, or Chapter 7 for CP/M discs) that there is nothing there that you want to keep. Before using disckit, remember that if you only want to copy a single file amongst several on a disc, this can be done within LocoScript (see Chapter 4)

or, for a CP/M file, using the CP/M utility **PIP** (Chapter 7).

If the destination disc is completely blank, diskit will detect this and perform **formatting while copying**. However some brand new discs have a magnetic pattern on them which is part of the manufacturer's test procedures. There is a danger that the Amstrad might confuse this for a formatted disc, and then get lost halfway through copying. So the safest thing to do is always to format new discs before use.

1. With One Disc Drive Fitted

To copy the entire contents of a side in diskit, press <f5> and the box on the screen will clear showing just one key, the <Y> key. This is used to confirm that you want to copy.

Place the **source** disc in the disc drive before pressing <Y>. The very early version (1.0) of the diskit program was rather vague here and didn't make it clear which disc should be inserted first. Later versions do tell you what is going to happen next. With the 8256, the copying operation is completed in two stages, the first 20 tracks are read from the source disc, then you are asked to put the **destination** disc in the drive for those tracks to be written. Then the discs are changed again and the operation is repeated for the remaining tracks. The number of the track being currently read or written is shown in the top left hand corner of the screen. Diskit uses the words **read** and **write** to describe its operations. It **reads** information from the source disc, and **writes** it to the destination disc. To be on the safe side, you can open the 'write-protect' hole on the appro-

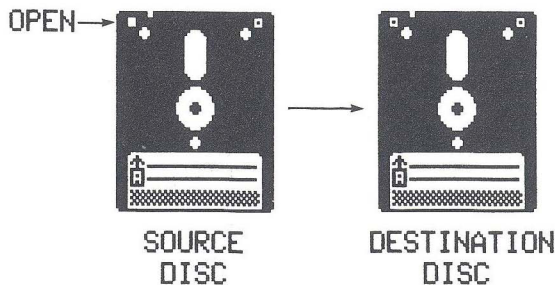


Figure 3.4 Write Protecting the Source Disc

prate side of the source disc before starting to copy to protect the contents from accidental corruption (see Figure 3.4). If you are copying a working disc (to make a back-up copy of all the files on it for instance), then don't forget to close the hole again before trying to use the disc again for creating or editing another file.

The computer will start to copy the contents of the disc into its memory. You can see the progress that it is making as the number of the track currently being read appears in the top left hand corner of the screen.

2. With Two Disc Drives Fitted

If you have two disc drives fitted, there are some choices. First you get a menu which asks which drive you wish to **read** from, then another menu asking which drive you want to **write** to. The choices are:

- to copy the contents of a CF2 disc onto another CF2 disc by swapping the discs over in drive A;
- to copy the contents of a CF2DD disc onto another CF2DD disc by swapping the discs over in drive B;
- to copy the contents of a CF2 disc in drive B onto another CF2 disc in drive A.

Copying a CF2 disc from drive B to drive A takes about one minute, and once started needs no intervention until completed. Do make sure that you have got the disc to be copied in drive B, or else you will end up with two empty discs. To be sure, you can write protect the source disc by opening up the write-protect hole; both surfaces of a CF2-D disc can be protected by opening just the write-protect hole on side A.

It's not possible with diskkit to copy from one sort of disc to another. This can be done either in LocoScript or in CP/M using PIP.

Copying Discs of Other Amstrad Formats

Although the PCW uses the same 3-inch compact-floppies as other Amstrad computers, the disc format isn't identical. The version of diskkit supplied with the PCW won't copy other Amstrad formats,

so if you have CP/M software, or BASIC programs on an Amstrad 6128 disc, then these can be copied using the PIP command under CP/M (Chapter 7). Discs created on a PCW can't be read by an Amstrad 6128 because the disc directory is in a different place.

<f1> Verify

The final utility available in disckit verifies the contents of a disc. This confirms that the contents can be read by the computer, and can be used at any time to check a disc that you suspect to be faulty. The program reads through the disc track by track, and if any reading errors are found, the program reports them, together with the name of the affected file (if any).

START OF DAY DISC

Eventually, **any** disc is liable to failure without warning. All the software used with the word processor is supplied on disc, so make at least one back-up copy of all the discs as soon as possible because the machine is useless without it. You will need to load the software into the computer each time that you switch it on (or change from LocoScript to CP/M and *vice versa*). So you will need a disc holding the software files for use at the start of each day. This is known as the **start of day disc**. Once LocoScript is loaded into the Amstrad's memory, then that disc can be removed and discs containing just document files can be loaded into the disc drive instead.

The disc supplied with the machine holding the LocoScript word processing program doesn't have much room left on it. It contains a lot of sample files which you aren't likely to need often, if ever again. You can delete these on the start of day disc to make more room for your own documents.

Here's how to do it. Make a back-up copy of the LocoScript disc using **disckit**. Then with the *new* copy in the disc drive, **boot it up** (computer jargon meaning *load the program into the computer and make it run*) by pressing the <SHIFT>, <EXTRA> and <EXIT> keys. Once you are in the disc management menu page, you will be able to use the cursor keys and then the function key <f6> to erase unnecessary files.

The files that you are unlikely to need on your *start of day* disc are:

PHRASES.STD
READ.ME (after you've read it of course)
ADVERT.EG
DOCUMENT.EG
LAYOUT.EG
QUOTE.EG
TEXT.EG1

These are all **text** files, that is files which can be edited and printed out under LocoScript. You can see what's in them by **editing** them as described in the LocoScript manual, or in Chapter 5. You probably won't need to keep all the examples of the template group on your start of day disc, provided that you keep them somewhere for reference. You should make your own templates to serve *your* exact requirements. Erasing all these unwanted files will liberate an extra 39 kilobytes of storage area.

All the remaining, hidden files are **program** files. To see them you will need to press <f8> (**Options**) and select **show hidden files** (move the cursor down the menu to this option using the <↓> key and then press the <+> key to the left of the <SPACE> bar, then press <ENTER>). The disc management menu will now show the software files, followed by the letter **h**. They are:

J20LOCO.EMS *
MATRIX.STD
SCRIPT.JOY
MAIL232.COM
(* earlier versions will have a different number)

The first three are the LocoScript software. The last one can be removed from your start of day disc. That is MAIL232, the communications program about which I will have a lot more to say in Chapter 11. This can't be used under LocoScript, only under CP/M, so if you are going to use it then you will need to transfer it to a CP/M disc, or even better a dedicated communications disc.

Do remember that these files should only be erased from your copy disc, not the one supplied with the machine.

DELETING UNWANTED FILES

Place the new LocoScript copy disc into the disc drive and make it run. LocoScript files are selected by moving the **cursor**, that is the highlighted strip, to cover the file name on the **disc management menu**. You can move this using the <↓> key (and if necessary the <←>, <→> and <↑> keys) until the cursor covers the name of the file that you want to erase. Then press <f6> (to get <f6>, hold down the <SHIFT> key and press the <f5/f6> key).

Now, you will see a pull-down menu showing the name of the file to be deleted. This is to double check that you really want to erase the file covered by the cursor. To confirm erasure press the <ENTER> key; if you have changed your mind, press <CAnceL>.

RECOVERING DELETED FILES

If you have erased a file by accident, all is not necessarily lost. Erased files are not deleted immediately, but moved to a category called **LIMBO**, which doesn't show on the menu unless you want it to. Eventually of course, the old file will be over-recorded with new material, but until this happens it can be recovered. To find out how to do this, as you are erasing the unwanted files on your LocoScript back-up copy, try to recover one of them. Put that disc into the disc drive and enter the disc management menu (press <f1> if you have just changed discs). Place the cursor over **READ.ME** (for instance) in the letters group. Now press <f6> by holding down <SHIFT> and pressing <f5/f6>. A pull-down menu will appear showing the specification of the file to be erased. If this is correct press <ENTER> to confirm the erasure. The disc drive will whirr and the **READ.ME** entry will disappear from the disc management menu. But note that at the top of the **LETTERS** column it now reads '**1 limbo files**' (not very grammatical, but we'll let that pass). This shows that the file has not been completely erased, but consigned to 'limbo'.

Now hold down <SHIFT> and press <f7/f8>, and a pull-down menu will appear. Using the down cursor key, position the cursor over **reveal limbo files** and press the <+> key. Now the menu will show:

READ.ME limbo

To recover the file again, place the cursor over the limbo file to be recovered. Now press <f5> for **Rename**. A pull-down menu invites you to **rename file**, **recover from limbo**, **rename group** or **rename disc**. You want to recover a file from limbo, so using the <↓> key position the cursor over **recover from limbo**, and press <ENTER>. Another pull-down menu now shows you which limbo file is being recovered and asks you to enter the name that it is to be called. Note that the file specification must be different from any other file on the disc, so a file-name mustn't be the same as any other in the same group. If you *do* enter a name already in use, the word processor will report an error and not allow you to continue until the name has been changed. Similarly only one limbo file of a given name is allowed. If you erase a file (thus sending it to limbo), and a limbo file of the same name already exists, then the old limbo file will be irretrievably lost and the new limbo file will take its place.

DISC NAME

You can give the disc a name. It will appear on the disc management menu page next to the disc drive number. To change the name of a disc, use <f5> (**Rename**) in the disc management menu. This will bring a pull-down menu. Using the <↓> key, choose the option *rename disc*. Another menu will show the present name (if any), and there is a space to fill in the new name. Confirm with the <ENTER> key.

GROUP NAMES

Each disc can hold files in up to eight different groups. The different groups aren't kept in any special places on the disc, but any file must be assigned to a particular group. If you format a blank disc, when you first look at the contents under LocoScript, the groups will default to:

group 0	group 4
group 1	group 5
group 2	group 6
group 3	group 7

The advantage of having *groups* on a disc is that, under Loco-

Script, you can set up *group templates*. These will save a lot of time and ensure consistency of layout between different documents of the same type. Templates will be explained in Chapter 4. To change the name of a group, go into the disc management menu, and place the cursor over a file in the group that you want to change the name of. If the group doesn't contain any files yet, you need to press <SHIFT> at the same time as the cursor control key to get the cursor over it.

Press <f5> (Rename), and a pull-down menu will appear. Using the <↓> key, choose the option **rename group**. Another menu will show the present group name (if any), and there is a space to fill in the new name. If the cursor *wasn't* on the group that you want to change, then type in the old group name and the new name in the appropriate places. If the group names are still the default ones as shown above, then it's only necessary to type in the number, and not the word **group**. Check that the **drive** is correct on the menu. If any detail is not correct, either the computer will refuse to rename the group, or else the group will be renamed incorrectly. Confirm with the <ENTER> key. The group will now be renamed on the disc.

4 Creating LocoScript Documents

INTRODUCTION

Word processing is all about creating documents. These will consist of words and characters which are entered into the word processor through the keyboard. The great advantage of the word processor is that you are able to arrange, re-arrange, add and delete characters until you are satisfied with the results.

LocoScript saves the documents onto files on its disc drives. Although it's usual to save onto the floppy disc, it's quite possible to save onto the memory disc in exactly the same way. There may be good reasons for doing this, but care should be taken that any such files are copied to the floppy disc before the Amstrad is switched off or you leave LocoScript, or else the file will be lost.

When using LocoScript you will be in one of two screens. These are the **disc management menu** (see Figure 4.1) and the **editing screen** (see Figure 4.2). At the top of each of these are the **status lines** which will give you a lot of information about what is going on. If you find yourself with a problem, study these, they'll probably tell you what is wrong.

EDITING SCREEN STATUS LINES

At the top of the editing screen are the status lines. These are important, particularly if you prefer not to have all the codes and effectors on the screen while you are editing. The top line tells you the drive, group and file-name of the chapter that you are working on, whether the printer is in use, and which of the drives is currently in use. The second line tells you what is going on **at the cursor position**, which


```

BOOK/CHAPTER 4      Editing text.      Printer idle. Using B:
Layout 1 +PIPS +LS2 +LP6      Page 1 line 9 of 58
f1=Show f2=Layout f3=Emphasis f4=Style f5=Lines f6=Pages f7=Modes f8=Blocks EXII
-----
(*Bold)
(*PitchPSb)Chapter.Four
(*Pitch10)Creating.LocoScript.documents. (-Pitch)
-----
(-Bold) (*Layout1)
-----
wordprocessing.is.all.about.creating.documents..These.will.consist.of.
words.and.characters.which.are.entered.into.the.wordprocessor. through.
the.keyboard..The.great.advantage.of.the.word.processor.is.that.you.are.
able.to.arrange,.re-arrange,.add.to.and.delete.characters.until.you.are.
satisfied.with.the.results.
LocoScript.saves.the.documents.onto.files,.on.its.disc.drives..Although.
it's.usual.to.save.onto.the.floppy.disc,.it's.quite.possible.to.save.onto.
the.memory.disc.in.exactly.the.same.way..There.may.be.good.reasons.for.
doing.this,.but.care.should.be.taken.that.any.such.files.are.copied.to.
the.floppy.disc.before.the. Amstrad. is. switched. off. or. you. leave.
LocoScript,.or.else.the.file.will.be.lost.
When.using.LocoScript.you.will.be.in.one.of.two.screens..These.are.the.
(*Bold)disc.management.menu(-Bold).and.the.(*Bold)editing.screen(-Bold)..At.the.
are.the.(*Bold)status.lines(-Bold).which.will.give.you.a.lot.of.information.abou.
is.going.on..If.you.find.yourself.with.a.problem,.study.these,.they'll.
probably.tell.you.what.is.wrong.
(*Bold)Editing.screen.status.lines(-Bold).

```

Figure 4.2 Editing Screen

little persistence and practice driving becomes second nature. At least you're not going to crash your Amstrad, and what may seem very complicated at the start is really a very user-friendly word processor.

CREATING AND EDITING DOCUMENTS

When you press <C> from the disc management menu to create a new document, LocoScript opens a new file on the chosen disc (floppy or memory).

LocoScript Files

LocoScript stores the documents that you create onto the floppy discs in files. These can be retrieved later for updating and printing as required. These files contain all the characters and control codes in the form of magnetically encoded numbers. There is more than just the text in a LocoScript file. At the beginning of each file,

LocoScript stores a **file header**, which is a block of information containing details of the layouts, tabs, identifying text, page header and footers, and details of whether the codes, etc are shown on the screen. All this takes up quite a lot of room on the disc, especially if there are a lot of different layouts and tabs. Each file takes up a minimum of one kilobyte of disc space (two kilobytes in the second disc drive), even if there is only the file header and no text!

LocoScript can create files other than proper LocoScript ones. **Blocks** (page 72) are essentially files which contain text, but no file header. **ASCII** files (page 167) are similar, and only contain characters within the standard ASCII range. If you try to load one of these files into the Amstrad simply by pressing <E> in the disc management menu then you will get the message **Not a LocoScript file**. This is because LocoScript needs the information contained in the file header, and can't work without it. However these types of file *can* be loaded if need be (see page 75).

Amstrad Disc Files

To understand better what you are doing during disc management, it helps to know how files are held by the Amstrad.

All the files on a disc are classified into **groups**. Files in one group are not held together physically on the disc; indeed, even one file can be spread over different tracks and sectors throughout a disc. A file is assigned to a group. If the file is moved from one group to another, all that happens is that the assignment is altered. If a file is erased, it's not wiped from the disc. First it is assigned to a sub-group called **limbo**, and the reference to it is removed from the directory (although *limbo files* can be shown if required). Eventually, when the disc space is required for new files, the sectors occupied by limbo files are used up.

Once a new file is open it consists only of the header. Now text entered at the keyboard is inserted into the file. Finally when you press <EXIT> to finish editing, the header and all the text you have typed in are saved to the disc file.

When you edit an existing document, LocoScript opens the old file, *and* a new file on the disc. It then copies the header from the old

file into the new one and displays the start of the old text on the screen. As you move down the document, it is progressively copied into the new file, together with any additions or alterations that you make. For this reason, when you edit a file, you must be sure that there is enough space on the disc to accommodate the old and the new files at the same time. Only when you finish editing does LocoScript close the old file and assign it to limbo. Up to that point, you could abandon editing and leave the old file intact. This is a very safe way of organising things because the old version at least is safe until you have finally finished with it.

Document Creation

Before you begin to create a file in LocoScript there are some things that you should check first. Here's a brief list of them.

CHECKLIST FOR CREATING A DOCUMENT

enough room on disc?	page 69
correct group?	page 55
correct file name?	page 56
group template ok?	page 57

Place your **start of day** disc into the disc drive and load up LocoScript.

Groups

Look at the disc management screen. You will see that each disc (including the memory disc) can hold files in up to eight different groups. For a disc in the upper disc drive (drive A) these are shown in the box in the upper left hand corner of the screen. Figure 4.3 shows the groups on the LocoScript disc supplied with the computer.

Figure 4.4 shows the same part of the screen with a newly formatted but empty disc in the drive. (If you change the disc in the drive you need to press <fl> to see the directory.)

You can set up your own groups on the disc. You don't have to use up all eight groups, but each group that you *do* use can hold a separate standard template for the documents that you create.

Drive A:			
127k used	46k free	25 files	
ADVERTISE	97k	group 4	0k
SAMPLES	12k	group 5	0k
CONT	1k	group 6	0k
TEMPLATE	17k	group 7	0k

Figure 4.3 Drive A Groups Menu (LocoScript Disc)

Drive A:			
0k used	173k free	0 files	
group 0	0k	group 4	0k
group 1	0k	group 5	0k
group 2	0k	group 6	0k
group 3	0k	group 7	0k

Figure 4.4 Drive A Groups Menu (Blank Disc)

CONTROLLING LOCOSCRIPT

LocoScript allows you to use it with the minimum amount of typing. There are many dedicated keys to perform special functions, such as moving through the text, etc. But the keys that you will use most frequently are the **function keys** <f1> - <f8>, the cursor control keys <←>, <→>, <↑> and <↓>, the <+> and <-> keys and <ENTER>.

Function Keys

These are the four keys in a vertical row immediately to the right of the <RETURN> key. The actual functions performed by these keys change for different LocoScript screens. The current functions are always shown along the third status line at the top of the screen. As you might expect, to get the even numbered functions, you press the appropriate key plus <SHIFT>.

Cursor Control Keys

These are shown in this book as <←>, etc. They are used to move the cursor, either the editing cursor around the text while editing, or the disc management cursor around the file-names in the disc management menu. These can also be used in conjunction with the <SHIFT> key for bigger jumps.

In the disc management menu the cursor keys plus <SHIFT> can be used to force the cursor into groups which don't yet contain any files.

In the editing screen, there are also dedicated cursor movement keys such as <PAGE>, which moves the cursor to the first character on the next page, and <DOC>, which moves the cursor to the bottom of the document. <EOL> moves the cursor to the end of the current line. All these will make the cursor move faster than holding down one of the cursor control keys. In the latter case the Amstrad has to keep checking the keyboard to see if the key is still being held down, and this wastes time. All these keys can be used in conjunction with the <ALT> key, which will make the cursor move in the opposite direction.

<+> and <->

These keys, on either side of the <SPACE> bar (not to be confused with the + and - character keys on the top row), are used to switch on and off many of the LocoScript functions, and also the printer controls in both LocoScript and CP/M.

<ENTER>

This key is used to tell the Amstrad to accept an instruction that you have just given. If you have changed your mind, press the <CANcel> key instead.

CREATING A DOCUMENT

In the disc management menu, place the cursor over the group that you want the new file stored in. If the group is on a different disc, change the discs over and press <fl>. If you haven't yet looked at *standard templates* (and you probably haven't because they're not explained until later), for now place the cursor over the **letters** group

if you are using single sheets of paper to print out, or over the **cont.** group if you are using continuous, computer-type stationery. Then press the **<C>** key. Don't press **<ENTER>** yet. After a moment you will see a *pull-down menu* with a description on it of the file that you are going to create. It shows the **file-name** and the **group**. The first thing to notice is that LocoScript has already given the file a name – **DOCUMENT.000**. If there is already a **DOCUMENT.000** in that group on that disc then the number in the file-name extension will be automatically incremented to **DOCUMENT.001**, **DOCUMENT.002**, etc. This is because only one file of a given name can be held in one group. LocoScript calls all new documents by this name, because it can't store a file without a name. But only one file of a particular name can be stored in any group at one time. If you store another document of the same name, the old one will be erased. So it's important that you give the file its own name, of your choosing. Choose the name carefully. Not only will it allow you to store the file safely, but it should help you to find it again later. To change the name on the menu, simply type over the old one. The file-name is in two parts, eight characters followed by a full stop, then three more characters. The second group is usually known as the *file-name extension*. The characters can be any letters or numbers, but not symbols. It really doesn't matter to the Amstrad what names you choose to call your files by, provided that there aren't two the same in the same group. Neither part of the name needs its full complement of characters; if you type a **<SPACE>** before the eighth character, the cursor jumps to the first position after the full stop.

Check also that the correct **group** is shown on the menu. If the group that you want the file to be in doesn't already have any files in it, then in order to be able to place the cursor over it on the disc management menu you'll need to press the **<SHIFT>** key as well as the cursor key. You can also change the group name in the appropriate space on the pull-down menu simply by typing in the new name. When all the details on the pull-down menu are correct, press the **<ENTER>** key.

STANDARD TEMPLATES

As you use your word processor more and more you will begin to appreciate the possibilities that exist for controlling the layout of the final document. Once you have experimented a little you will see the

sort of material that you regularly create. In all probability, the number of different layouts that you'll need come down to a few fairly standard (to you) formats. Then you'll appreciate the value of setting up some **standard templates** to save time as you set up to create each new document. A standard template is simply a document that is already prepared and stored in a particular group on the disc. The computer automatically copies this into its memory when you press <C> to start to create a new document in that group. This can save you time and effort, as well as making life a lot simpler in the long run. Once you have a standard template, all the documents created in the group will be of uniform layout, without any further effort on your part. The template can be as simple as just the left and right hand margins, or as complicated as a whole standard acknowledgement letter, with spaces for you to type in the date and the name and address of the recipient. The important thing is that this template must be stored as a file in the group under the name **TEMPLATE.STD**.

Creating a Standard Template

A good place to start experimenting is with the template provided in the **letters** group of the LocoScript disc. Using your back-up copy of the LocoScript disc, place the disc management cursor over **TEMPLATE.STD** in the **LETTERS** group on disc A, then press <E> to edit that file. A pull-down menu will show you the file that you have selected. If this is really the one that you want, press <ENTER> to confirm your choice. By editing the standard template, the new version will be saved in the place of the old one. The old one will be consigned to limbo, and then eventually lost if not recovered. The template you are going to edit appears on the screen as shown in Figure 4.5. The second line of the top highlighted section shows how the layout is set up.



Figure 4.5 Standard Template - Layout Strip

Layout shows that this is the **base layout** (if another layout has been selected this would show *Layout 1* or whatever).

- Pitch 12** shows that the letters are in 12 pitch, that is 12 letters to the inch.
- Line Pitch 6** shows that the line pitch is 6, that is 6 lines will be printed out every inch down the page.
- Line Space 1** shows that the line spacing is 1.

To alter any of these parameters in the base layout you select **<f7> = Modes**. This gives a pull-down menu with four choices:

Edit Header
Edit Identify Text
Insert Text
Disc Management

There is the possibility of confusion here, because LocoScript has two different sorts of **header**: there is the **file header**, which is information recorded at the beginning of every LocoScript file, and there are **page headers** – text which can be printed automatically at the top of each page. The page headers are recorded as part of the file headers. Confusing, isn't it? Never mind, it'll all come clear in the end. At this moment we want to edit the *file* header.

Look again at the pull-down menu that has just appeared. The menu cursor is already at **Edit Header** so, as this is what you want to do, simply press **<ENTER>**. The screen will clear and you will see four horizontal bars with the legends **end of header**, **end of footer**, etc. You need not bother with these now. The status line at the top of the screen showing the f-key functions has changed. Press **<f7>** (Options). The screen clears again and the function key status line changes once more. This time you want **<fl>** (**Layout**).

Now the top line of the screen changes again to read:

Editing base layout.

The second line shows us the state of the **letter pitch**, **line pitch**, **line space**, **italics** and **justify**. Press the **<->** key and you will see the cursor move between these. The **<->** key moves the cursor back again. These settings can be altered using the **<+>** and **<->** keys which are either side of the **<SPACE>** bar. Move the cursor back to cover **letter pitch**.

Letter Pitch

With **letter pitch** highlighted, press the <+> and <-> keys and watch the values change. They are:

- PS - proportional spacing, where each different letter, instead of occupying an equal amount of space as with a typewriter, only occupies a space proportional to its width. The letters in proportional spacing are approximately the same size as those of 12 pitch (see below).
- PS D proportional spacing with double width characters, that is, each character twice as wide as usual.
- 10 10 pitch, that is 10 letters to the inch, where each letter irrespective of its actual width occupies a space of equal width ($\frac{1}{10}$ th inch).
- 10D 10 pitch double width, that is 5 letters per inch.
- 12 12 pitch letters – equal spacing – 12 per inch.
- 12D 12 pitch double width characters – 6 per inch.
- 15 15 pitch letters – a plainer style than PS, 10 or 12 pitch without serifs.
- 15D 15 pitch double width characters – $7\frac{1}{2}$ per inch.
- 17 17 pitch letters – similar to 15 pitch.
- 17D 17 pitch double width characters – $8\frac{1}{2}$ per inch.

See the examples shown in Figure 4.6.

Line Pitch

Next, using the <-> key, move the highlight to **Line Pitch**. The two possible settings for this are 6 or 8. This means that either six or eight lines are printed down every inch of the paper. The characters will be the same size, but at 8 pitch they will be closer together than at 6 pitch. The choice is made using the <+> and <-> keys.

Line Space

The next setting to make is the Line Space, which is the number of line pitches between each line. The chosen value can be 0 (the paper

```

10 pitch letters
10 pitch double
12 pitch letters
12 pitch double
15 pitch letters
15 pitch double
17 pitch letters
17 pitch double
Proportional spacing
PS double width.
Italic lettering.

```

Figure 4.6 Letter Pitches

doesn't move on between lines so that they are all printed on top of each other), through 1, 1½, 2, 2½ to 3 (where the paper moves up three spaces before the next line is printed). The <+> key is used to increase the value, the <-> key to decrease.

Italic

The next setting is *Italic*. Use the <+> and <-> keys to switch this on or off for italic (sloping letters) as required. If *Italic* is selected, a tick will appear beside the word.

Justify

The final choice on the top line, again reached using the <-> key, is **Justify**, where the right hand as well as the left hand edges of the text are lined up, padding out short lines with extra spaces as necessary. The word processor does this extremely well in the final printed out version of the text, although it may not always appear so on the screen. Only fixed spacing is available on the screen, so if the pitch chosen is proportional spacing, lines containing several 'short' letters (on paper) will appear longer on the screen. If *codes* are inserted in the text, they won't make the *printed* lines longer, but if they're shown on the screen they'll make the line longer there. These can be removed from pre-viewing using <f1> and removing the top item on the **show menu** presented using the <-> key followed by <ENTER> (see **show** page 71).

If you change your mind about any of the choices that you have made, you can go back and forward along the line using the <←> and <→> keys, and make alterations with the <+> and <-> keys. You can even come back to re-edit the header at a later time.

Ruler

Once you're satisfied with the setting on the top line, use the <↓> key to move the cursor to the ruler on the lower line. Here, using the <←> and <→> keys you can move it left or right along the ruler. On the ruler you will see the left and right hand margins as they are pre-set, and also some tabs, shown by →s and ↔s. You can alter these as you wish.

To see exactly what you can do, first of all remove all the tabs that have been put in. Move the cursor to the extreme left hand point on the screen over the 0 on the scale. Then press <f1>. This sets the left hand margin to zero. Now move the cursor to cover each tab setting in turn and press the <-> key to erase them. If you keep on pressing the <→> key, the image on the screen will scroll off to the left, and you will be able to continue on along the scale until you come to 255. In theory you could create a document this wide, but using the standard printer supplied with the Amstrad word processor the widest that you could print, even using 17 pitch letters, would be about 170 characters wide. The main reason for being able to accommodate this number of characters across the screen is that when using **show**, all the codes are displayed, so a lot of extra space may be required in any one line.

Left Margin

Now move the cursor to the point where you would like the left margin to be set. If you don't have a particular opinion on this yet, try moving the cursor to 10 (number 1 on the scale). Then press <f1> and the **left margin** will be set at that point.

Right Margin

Now move the cursor along the ruler to the position where you would like the **right margin** to be set and press <f2>.

Tabs

Tabs (tabulations) are pre-set points across the layout to help you to lay out your text in neat columns. LocoScript has four different sorts of tabs, but you may only need to use one or two types. The normal tabs are just like those on a typewriter. They are shown on the layout strip as →. When you press the <TAB> key during editing, a → appears on the screen, and when printed out the next character will line up with the next pre-set tab position (as shown in Figure 4.7). Fixing the tab positions is much the same as setting the margins. Move the cursor along the lower line of the layout strip until it is where you want the tab to be. Then press <f3> and you will see a → appear. To remove an unwanted tab, place the cursor over it again and press the <-> key.

```

B:BOOK/TABS.DEMO      Editing text.      Printer idle. Using E:
-Layout  -P10  -L51  -LP6      Page 1 line 1 of 54
f1=Show  f2=Layout  f3=Emphasis f4=Style f5=Lines f6=Pages f7=Modes f8=Blocks EXII
0.....
→ Ordinary.tabse
→ line up on
→ the left
→
→ → Right.tabse
→ → line up down the
→ → right hand side
→
→ Decimal points line up → 123.00
→ beneath decimal tabs → 4567
→ → → 23456
→
→ → → Lines arrange themselves
→ → → symmetrically
→ → → around a centre tab

```

```

Ordinary tabs
line up on
the left

```

```

Right tabs
line up down the
right hand side

```

```

Decimal points line up      123.00
beneath decimal tabs      4567
                          .23456

```

```

Lines arrange themselves
symmetrically
around a centre tab

```

Figure 4.7 Tabs

Right Tabs

The right tab is shown on the screen as a ←. With this, when the <TAB> key is pressed, the next block of characters will line up with *the right hand character* lining up with the tab mark. Of course, the text must not be so long that it is forced past the tab mark.

Centre Tabs

This is shown as a double arrow (↔) on the layout strip. After the tab, any text typed in is *centred* around this point.

Decimal Tabs

Shown on the layout strip as an asterisk (*), this is used to line up columns of figures. For figures without a decimal point this acts in the same way as a right tab, but if there is a decimal point, then the point is always kept under the decimal tab. When this is used with columns of numbers, all the units, tens, hundreds, etc line up correctly in columns.

When you are satisfied with the settings for the base layout, press the <EXIT> key to get back to the **Editing header** page. These settings need not necessarily be final, you can go back to change the base layout at any time, although LocoScript will scroll back to the beginning of the document to do so.

New Layouts

All the features described above can be changed in the middle of a document by creating **new layouts**. These can be defined by pressing <f2>=**Layout** from within the editing screen. A pull-down menu invites you to create a new layout, or edit an existing one. If you choose to create a layout, you can decide what it should look like in the same way as with the base layout, but you can see the effects much more quickly and go back to adjust them until you are satisfied. You can store several different layouts in the header, up to 99 in fact, but as each one takes up header space, you need to declare how many you need. This is done by editing the header. Press <f7>=**Modes**, followed by <f7>=**Options**, then <f5>=**Tab count**. A pull-down menu allows you to enter the maximum number of layouts and the maximum number of tabs in each layout. The nor-

mal pre-set number of each is 5 layouts and 10 tabs in each. The theoretical maximum number possible is 99, but this would require an enormous header occupying **11 kilobytes** of disc space even if there were no text. If you find that you run out of layouts or tabs in the middle of a document you can always go back and increase the number in the file header as required.

Page Length

From the editing header page, press <f7>=**Page Length**. This value represents the length of each page of the document in single lines of six lines per inch pitch. A single sheet of A4 paper is $11\frac{2}{3}$ inches long, so the page length should be 70 lines. For standard $9\frac{1}{2}$ inch **continuous** computer stationery, the page length is 11 inches, or 66 lines. This is the *total distance* between the start of one page and the start of the next. **It's important that the page length set here is the same as the page length set in the *printer parameters*** (see Chapter 6).

Figure 4.8 shows a typical page layout.

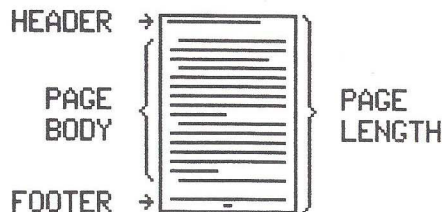


Figure 4.8 Page Layout

LocoScript allows us to format the page in other ways too.

Headers and Footers

When you have finished editing the layout and pagination, press <EXIT> and you will return to the screen for headers and footers. In this case, header refers to **page headers**. When each page is printed out it can have a header and footer. These are areas at the top and bottom of each page which are automatically printed with a standardised text. LocoScript allows some variation in the headers and footers, so that for instance, the first page can have a different head-

er, or the last page can have a different footer. Headers and footers are recorded in the file header. Unfortunately it's not possible to view the final effect of the pages combined with the headers and footers on the screen, you must wait until they are printed out on paper. You can decide whether the headers and footers are standard for all documents in a group, or else change them for each individual document. A compromise is to create *generalised* headers and footers in the group standard template, which need minimal modification as each new document is created.

If you are creating a document of *more* than one page length you may wish to print a standard text at the top and bottom of each page. This can be different on the first page, or the last page from the rest of the document, or it can be different for odd and even numbered pages (as with the left and right hand pages in a book).

In the **pages menu** you must define the size of the zones to accommodate the headers and footers, and the positions within those zones where the headers and footers are to be printed.

Header Zone

Enter here the number of lines of space that you want at the top of each page before the main text starts. If you wish, the header space can be left blank.

Header Position

If you *do* want to print a header, this is the line on which it will start. This line must be within the header zone, and if the header itself is of more than one line, then there must be enough room for it before the end of the header zone.

Footer Zone

This is the number of lines between the bottom of the main text on each page, and the bottom of the sheet of paper. If you wish, you can print a footer in this space.

Footer Position

If you decide to print a footer, then this is the line on which it starts.

then a single number would be centred in the position, and if you were to use:

<<

the single figure would be to the left of the position.

If you still have version 1.0 of LocoScript, don't waste time trying to make this work, it doesn't.

You can also show the total number of pages in the document, for example:

Page No. (PageNo) >> of (LPageNo) >>.

This will show the total number of pages in that document. To get last page number enter:

<+>LPN<ENTER>

If you have broken a long document up into shorter sections (chapters for example), then you can set the first page number for a particular section. This is done in the **pagination** pull-down menu. From the editing screen you get there by pressing <f7>=**Modes** then <f7>=**Options** then <f8>=**Pagination**.

Pagination

The pagination menu allows you to choose the page number at which page numbering starts, and also allows considerable choice in the way that the headers and footers are printed out.

The simplest choice is for all pages to be the same, that is just one standard header and one standard footer printed out on each page throughout the document. However, there are some circumstances where a different header and/or footer is required on some pages. For instance, in a memo you might want all the pages except the last to carry a footer which says:

/continued

This would be inappropriate on the last page, so you can choose either to disable the footer on the last page, or else print a different one, for example:

/Ends.

If the first page carries a prominent heading, you may not wish to print a header as well. This can be disabled, or a different header chosen. If you are preparing the pages for a booklet which is to be printed on both sides of the paper, then you might want a different

header and footer for odd and even pages so that the page numbers are correctly positioned on the outer corners of the pages. This too is possible. Two different sets of headers and footers are allowed in any one document. If the settings in the pagination menu call for two sets, then when you get back to the screen for entering the header and footer text, you will see the legends below the spaces for the text labelled appropriately.

Pressing <EXIT> again causes a pull-down menu which asks if you are satisfied with the settings. Press <ENTER> and you are back in the editing screen. You can now enter text, or save the header to disc to create a standard template.

Saving a Standard Template

Once you have finished creating the header you should save it in a group on the disc as **TEMPLATE.STD** by pressing <EXIT> and choose **Finish editing**. Next time you create a new document in that group LocoScript will automatically use it to create the header for the new file. Of course you may not want that exact file header on every document, you may want different page headers and footers for example. But making slight alterations to the file header is much less trouble than starting from scratch each time. However, changes made to the header of a document will affect *only that document*. If you want to use the layout for all future documents in that group, you will have to change **TEMPLATE.STD**. An easy way may be to make a copy of the document, delete all the text and rename what's left **TEMPLATE.STD**. Make sure that you have a back-up copy of the template on another disc.

Your LocoScript disc as supplied with the machine has a number of example templates. Having tried out the various functions, you'll now be able to adapt any of those to suit your own needs.

Once you start to create a new file, the Amstrad has opened an empty file (with a header copied from the standard template of the group that you are in) for you to fill with text. This is now essentially the same as editing an existing file, which is covered in the next chapter.

5

Editing LocoScript Documents

EDITING EXISTING DOCUMENTS

Remember that when you want to edit a document, you must have enough room left on the disc for the old file and the new file at the same time. You can check this on the disc management menu. Each file entry in the disc management menu shows the size of that file. The entries just below the status lines for drive and group information show you how much space has been used on each, and how much remains free. If there isn't as much room left on the drive as is needed for the existing file plus any additions that you are likely to make, you should copy the file to another disc and edit it on that. In an emergency you could copy the old file into the memory disc, edit it there and then transfer it back to the floppy-disc, erasing the old file first. But this isn't recommended because there is the possibility of losing both versions.

Checklist for editing a document

Enough room on disc for new version?

(check on disc management menu)

Is the old version required?

(make copy and edit that)

Check header

(title, version number, dates, etc)

To edit an existing document place the cursor over the appropriate file in the disc management menu and press <E>. A pull-down menu will show your choice for confirmation, which is done by pressing <ENTER>. If you are confident that the cursor was over the file that you wanted, pressing <ENTER> immediately after <E> saves a little time.

CODES AND EFFECTORS

Many different effects can be incorporated into your documents. The **base layout** will determine how the document will look if you make no further changes, however additional layouts can be incorporated as required, and localised changes can be made using **codes** and **effectors**. Words can be underlined or italicised. If you are not sure of the possibilities, when you are in the editing screen press the <+> key and shortly a pull-down menu will appear showing a list of effects. (These are listed below, and also in Appendix 1.) You can speed up the arrival of this menu by pressing the key in the middle of the cursor control keys, immediately after pressing <+>. If you move the menu cursor down with the <↓> key until you reach the effect that you want, and then press <ENTER> a **code** will be inserted into the text at the position of the editing cursor. The code is a special character that is inserted into the document. It isn't printed out, but it will affect the way that the document is printed out. The code can either be invisible on the editing screen or be seen, as you prefer (see **Show** later). Here is a list of the possible effects:

<u>Effect</u>	<u>Value</u>	<u>Abbreviation</u>
Bold		<+>B
Centre		<+>C
Double		<+>D
Italic		<+>I
Keep	??	<+>K
Layout	??	<+>L
Line Pitch	??	<+>LP
Line Spacing	??	<+>LS
Last Line		<+>LL
Last Page Number		<+>LPN
Pitch	?? ?	<+>P
Page Number		<+>PN
Reverse		<+>R
Right Justify		<+>RJ
SuBscript		<+>SB
SupErscripT		<+>SR
Underline		<+>UL
Word underline		<+>W
UniT		<+>UT

hard space	<+> <SPACE>
hard hyphen	---

The effect that you have selected will continue in the document until cancelled. This is also done by inserting a code into the text, this time preceded by the <-> key. Press the <-> key to see the menu.

<u>Effect</u>	<u>Value</u>	<u>Abbreviation</u>
Bold		<->B
Double		<->D
Italic		<->I
Keep	??	<->K
Layout		<->L
Line Pitch		<->LP
Line Spacing		<->LS
Pitch		<->P
Reverse		<->R
SuBscript		<->SB
SupeRscript		<->SR
UnderLine		<->UL
soft space		<-><SPACE>
soft hyphen		---

There is a quicker way to insert these codes, and that is to use the **abbreviations**. These are shown in the tables above, they are also shown in the pull-down menus as the *capital letters* (but you don't need to use capitals). Simply press the <+> or <-> key and then the abbreviation, followed by <ENTER>. If the abbreviation is unambiguous, the code is inserted straight away. If not, a shortened menu is presented for you to confirm the choice.

Effectors

These are **returns** and **tabs** which have an *effect* on the layout of the text.

Show

If you put codes and effectors into your text you need to be able to see where they are, from time to time at least, or else you may get some unexpected effects when printing out. On the other hand,

when the codes are shown on the screen, it's difficult to see how the text is laid out. I like to see all the codes all the time except when checking the layout. I know other people who like a completely clear screen except when checking the codes. LocoScript caters for all tastes by allowing you to see what *you* want at any time. In the editing screen press <fl>=Show. A small pull-down menu allows you to choose whether or not you can see the state of:

Codes

Rulers

Blanks (empty spaces)

Spaces (when the <SPACE> bar was pressed)

Effectors

To see any of these, move the menu cursor down with the <↓> key and press <+>. Ticks in the menu against the names will indicate which are to be shown. The menu can be recalled at any time, and ticks removed using the <-> key. The choices made here when the file is saved will be those in effect when the file is next opened for editing.

PHRASES AND BLOCKS

Two very useful features of LocoScript are **phrases** and **blocks**. These allow you to store, and then subsequently re-insert with very few keystrokes, short phrases or longer blocks of text. This can save a great deal of typing, and can also help to ensure that difficult words are correctly spelt every time that they are used. Phrases and blocks are very easy to use once you get the hang of them and are well worth trying.

Phrases

If you want to copy or move a word or short phrase from one part of the text to another, then move the cursor to the first letter of the phrase. Press <COPY> and then move the cursor to the end of the phrase. The characters that you've moved over with the cursor will go into **reverse text** (that is, they will appear as black letters on a light green background) so that you can see them easily. When you come to the end of a line the cursor (and reverse text) won't wrap around to the next line automatically. If the phrase is over two lines then move the cursor using the <↓> key, and then use the <←> and <→> keys to bring the cursor to the end of the phrase. You have gone as

far as you need when the flashing cursor is on the character or space *after* the end of the phrase. The strip at the top of the screen now prompts you what to do. To *copy* the phrase, you press the <COPY> key again, followed by a letter key. In this case the highlighting disappears leaving the phrase in place, but it is now stored for subsequent use. To *move* the phrase somewhere else, press <CUT> and then one of the letter keys. The characters in reverse text disappear and are stored as a **phrase**.

The stored phrase can now be re-inserted into the text by moving the cursor to the point where it is to go and pressing <PASTE> followed by the designatory letter. It's possible to save up to 26 phrases in this way, each one assigned to a letter of the alphabet. Each phrase will stay in the Amstrad's memory until it's changed or until you leave LocoScript. The phrases can be saved to disc. If they are held in a file called **PHRASES.STD** they will be loaded automatically into the Amstrad's memory on *boot-up*. I reserve <Q> for temporary storage, but it's useful to develop a **PHRASES.STD** file as it can save a lot of typing. It's useful to keep a list of the phrases stored. This is described in **saving phrases** (page 74). There's a limitation on the maximum amount of text that can be stored in this way, 550 characters in all. Any phrase can be of any length, up to the limit, but the total length of all the phrases together must not exceed 550. However if you want to copy or move a longer passage there is no problem; instead of saving it as a phrase, you save it as a **block**.

Blocks

Blocks are saved in exactly the same way as phrases, except that they are assigned to **numbers** between 0 and 9 instead of letters. There is no limit to the length of a block in the way that there is with phrases. The blocks can be re-inserted into the text using <PASTE>. Blocks can be saved to disc to be moved to a different document, or saved for future use. For example, you might want to save a series of standard paragraphs or passages to be incorporated into letters or contracts that have to be frequently produced. This technique is known as *boiler-plating*.

MOVING TEXT AROUND

One of the most useful features of word processing is the ability to move text around a document without having to re-type it. This is

done on the Amstrad word processor using the **phrase** and **block** feature. To move some text from one part of a document to another you need to save it temporarily as a phrase or block, move the cursor to the position where you want it to go, and then insert it using the <PASTE> key. Here's how you do it. Simply move the cursor to the beginning of the text to be moved, press <COPY>, move the cursor to the end of the words and press <CUT> followed by a letter or number. This removes the section of text from its present position on the screen and stores it in the Amstrad's memory in a location marked by a letter or number. A single or very few words should be saved as a phrase using a letter, a longer passage as a block using a number. (I always reserve phrase <Q> for saving temporary phrases, it's a key that's convenient to remember and find). Then move the cursor to the point where the text is to go and press <PASTE> followed by the letter or number. If the text is to be moved to a different file, then it must be stored as a block and then saved to disc.

SAVING PHRASES AND BLOCKS

To save phrases and blocks for future use (or to move to a different file), press <f8> (**Blocks**). A pull-down menu appears showing the phrases and blocks stored in the memory. To save the phrases that are stored, use the <↓> key to move the cursor down to **Save all phrases**, and press the <ENTER> key. The phrases are saved, *but only to the memory disc*. Be sure to copy the PHRASES.STD file to the floppy disc before leaving LocoScript.

Blocks are saved individually, again using <f8> (**Blocks**). The pull-down menu shows the blocks currently stored in memory. Enter the number of the block to be saved to disc, and press <ENTER>. You will now be put into the **disc management menu**. Place the cursor over the group in which the block is to be saved and press <ENTER> again. Then you will be presented with a pull-down menu. Enter the file-name for the block. It's a good idea to use the file-name extension (ie the three characters after the full stop) to show that the file is a block and not an ordinary LocoScript file. So, for instance, if you had a set of standard business terms that you wanted to include in your letters, the file-name could be:

TERMS.BLK

When you press <ENTER> again the disc management menu will disappear and you will be back in the editing screen at the point where you left off.

INSERTING TEXT

Blocks are saved to disc without the normal LocoScript file header. If you try to edit a saved block you will get the error message:

Not a LocoScript file.

To insert a saved block into a document, place the cursor at the point where you want the text to be inserted and then press <f7> (**Modes**). From the pull-down menu, select the option **Insert text**. You are presented with the **disc management menu**. If you only have one disc drive and are editing a document that is on the floppy disc, you can't change discs in the middle, so make sure that any blocks that you want to use are either in the disc or in the memory disc. Choose the block to be inserted and press <ENTER>. You will return to the editing screen and the block will be inserted at the cursor position.

You aren't restricted to blocks when inserting text. If you specify an ordinary LocoScript file for insertion, the header of that file will be stripped off and the text will be inserted in the same way as a block. If the document that you are inserting contains layout codes, these will be interpreted as the layouts of the document into which it is being inserted. There may be some strange (although logical) effects if the tabs for one layout are inserted into another. If you want to turn a block into a document, simply create a new empty document and insert the block into that.

EXIT

When you press the <EXIT> key you are presented with a pull-down menu offering four options:

- Finish editing**
- Save and continue**
- Save and print**
- Abandon edit**

Finish Editing

This is the default option, it saves the current version of the file that you are creating or editing to the disc under the file-name and group as defined when you started. If there was a previous version (if you were editing an existing file), it is consigned to limbo. Any previous limbo version will be lost. LocoScript then goes to the disc management menu ready for the next task. If you weren't at the bottom of the file, then LocoScript will scroll through the entire document before saving.

Save and Continue

This is much the same as **Finish editing** except that LocoScript returns to the file so that editing or creating can continue. Use this if you are in the middle of working and you want to save what you have done so far, for safety. Annoyingly LocoScript returns to the start of the file, not the place where you left off. If you were at the bottom of the file, simply press <DOC>. If you were somewhere in the middle, note where you are on the **page** and **line number** display at the top right hand corner of the screen before pressing <ENTER>. Then press the <PAGE> key the appropriate number of times (one after another will do). You can do this straight away, the key presses will be queued to take effect once LocoScript is back in the editing screen. Present versions of LocoScript scroll through text in a rather leisurely way (later versions may be quicker), so queuing commands like this can leave you free to get on with important tasks like making a cup of tea!

Save and Print

This option is also much the same as *Finish editing*, but this time LocoScript goes straight on to print out the document. Make sure that the **printer parameters** are correct before you exit. It's probably safer to use **Finish editing**, check the printer parameters and then use **Direct printing**.

Abandon Edit

This quits creating or editing without saving the current version, and without affecting any version already existing on the disc. Use this if

you have loaded a file simply to check on the contents and haven't made any alterations.

BACK-UP COPIES

Finally, when creating documents, don't forget to make a back-up copy of any important files. Similarly, when you've finished editing an existing file, you should update the back-up copy too. LocoScript makes it very difficult to lose a file accidentally, but don't forget that any disc can fail and become unreadable. Copying files is very easy with LocoScript, requiring very few key depressions (that's computer jargon for *not much typing*).

Making Back-up Copies with One Disc Drive Fitted

In the **disc management** menu, make sure that the cursor is over the file to be copied and press <f3>. Then using the cursor keys move the cursor to any group on the memory disc. Press <ENTER> and a pull-down menu will show the old file-name, group and disc, and then the *new* file-name, group and disc. Any of these specifications can be changed using the cursor keys and typing in the required specification, but in all probability this won't be required. Then, when you're satisfied, press <ENTER> again to confirm. The cursor will be over the new copied file. Leave it there for the moment.

So far you've made a copy of the file on the **memory disc**. This is no good as a permanent copy because it will disappear the moment that the computer is switched off, or when you go out of LocoScript. But it does allow you to change the disc (there's not much point in keeping the back-up copy on the same disc as the original, because if there *is* a failure, the whole disc *could* become unreadable). When you have put a different (formatted) disc in the drive, press <f1> for **disc change**. When the menu has changed to show the contents of the new disc, check from the new menu that there is enough space on the disc for the file that you are copying (you will be able to see the free space on the disc *and* the length of the file you are copying from the menu). Check also that there isn't a file already in the group on the new disc with the same name as the one that you want to copy across. The computer won't allow two files of the same name in the same group on the same disc. If this is the case, either delete the

existing file on the disc (see page 45), change its name (see page 46), or change its group (see page 47). If you don't need to do any of these things, the highlighted strip will still be over the file copied onto the memory disc. Press <f3> and then move the cursor to the group on the new disc where you would like the back-up copy to be recorded. Press <ENTER>, check the specifications on the pull-down menu that has appeared and press <ENTER> again to confirm. The whole process is easier to do than to describe, and you now have a back-up copy which should make sure that you don't lose your precious file.

If you get a sudden crisis when finishing an edit, such as the message **Disc full** half way through saving to disc, the best advice is – *don't panic*. Press <f7>=**Modes** and choose **Disc management**. Provided that the new version of the file that you have been editing isn't enormously bigger than the previous one, you should be able to copy one of the files on the disc into the memory disc and temporarily erase the version on the floppy disc. Then press <EXIT> and the Amstrad will finish saving the new version of the file. Then there should be room on the compact disc to copy back the temporarily saved file in the memory file. Not recommended as standard procedure, but a way out of trouble.

Making Back-up Copies with Two Disc Drives Fitted

If you have a second disc drive, making back-up copies is even simpler. I usually edit files on **drive B** but make back-up copies onto **drive A**. This means that if a disc fails I have a back-up, and if a drive fails I also have a back-up (I harbour a profound mistrust of technology!). With the working disc in drive B, and back-up disc in drive A (press <f1> in the disc management menu if you have just changed discs) move the cursor to the file to be copied, press <f3>, move the cursor to the group on drive A where you want the back-up copy and press <ENTER>. If there is already a back-up copy in the group on drive A, you will either have to erase it first, or copy to a different group before erasing the other (this latter course of action is safer). If you want to make a back-up copy of a disc *originated* in **drive A**, which can also be *read* in **drive A**, then move the **source** disc to **drive B** where it can be *read* (but not *written* to), and the **destination** disc to **drive A**, then proceed as above.

6 Printing Out

INTRODUCTION

The Amstrad word processor comes complete with its own printer – a dot matrix printer. That is, the characters that it prints out are made up of dots, caused by a single vertical row of wires pushing the inked printer ribbon into contact with the paper. These make vertical rows of dots on the paper, the number of rows depending on the letter pitch in operation. There are two printing modes, **draft quality** and **high quality**. Draft quality is suitable for scanning the text to check the final layout and to make any necessary corrections. The print head makes a single pass over the paper when printing, and the printer in this mode is *bi-directional*. That is, every alternate line is printed with the print head moving from right to left. This saves a lot of time as the head has to make fewer unproductive passes over the paper. High quality print out has much better appearance, but for each line of text the print head makes two passes (to print twice as many dots). Each of these passes is from left to right so that the superimposition is more accurate, but makes this sort of output much slower. (Figure 6.1 shows the output from the two different printing modes.)

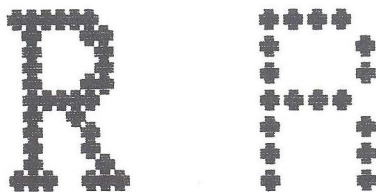


Figure 6.1 Dot Matrix Letters

But the time taken to print out a document need not be wasted. The Amstrad can be used to edit one document whilst another is being printed out. Both activities will be slowed down slightly, but not enough to cause problems. If the document that you want to edit is on a different disc to the one to be printed out, copy the file to be printed into the memory disc. Then you can change floppy discs and work on the new file whilst printing out directly from the memory disc. (Printing out is explained later.)

The actual formations of dots which make up each character are determined by the software. As with virtually all the software for the Amstrad, this is on disc, either within LocoScript or CP/M. For most purposes, the quality of the print is more than adequate.

It is possible to connect a different printer to the Amstrad, using the optional RS232/Centronics adaptor known as the **CPS8256** (also used for data communications, see Chapter 10). Using this, it's possible to connect most popular makes, however there is a snag. It's **not** possible to use the adaptor under present versions of LocoScript (up to 1.2). It's possible that later versions will allow this. It *can* be used under CP/M, and the most convenient way to use it would be to buy a CP/M word processing package such as **NewWord** or **WordStar**. Under LocoScript you will need to make an ACSII file of your document (see page 167) and then print that out using the **type.com** routine provided with CP/M (see page 98), but that's likely to be very tedious, particularly if the document is more than a page long. For a Centronics printer you will need a parallel lead as shown in Figure 6.2. For a serial printer you will need a null modem cable (see Figure 12.4 on page 176).

CONTROLLING THE PRINTER

Perhaps the one feature of the Amstrad that most people experience difficulty with is controlling the printer. This really isn't necessary, but probably stems from the fact that whereas with other computers and word processors the printer is a completely separate item, with the Amstrad it's all part of the one machine. Other printers have buttons on them for line feed, form feed, etc. With the Amstrad all this is controlled by keys on the keyboard. The keys are:

<PTR>, <EXIT>, <+>, <->, <↓>, <f1>, <f2>, <f3>, <f5>, <f7> and <f8>

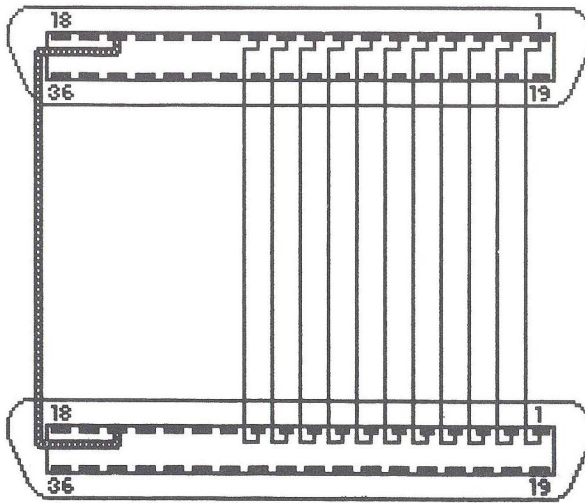


Figure 6.2 Centronics Printer Cable

<PTR> Key

The <PrinTeR> key switches the keyboard to control the printer. In this state all other operations cease until the <EXIT> key is pressed, when everything resumes where it left off. In the <PTR> control state, the status lines at the top of the screen, either in the disc management menu or the editing screen, change to show the state of the printer, and the appropriate function keys for printer control. As each function key is pressed, a pull-down menu appears with the various options (see Appendix 1). To get rid of the pull-down menu press the <ENTER> key, and to leave the printer control state press the <EXIT> key. A similar thing happens in CP/M except that the printer controls are shown along the bottom of the screen and there are no pull-down menus. Here are the functions of the various keys in LocoScript.

<f1>=Options

This key gives a pull-down menu which sets the major printer parameters. Before you start printing, you should be sure that these are set up to meet your requirements. The menu shows the options. Use the <↑>, <↓>, <+>, and <-> keys to alter them as you wish. Press <ENTER> when you are satisfied and the menu will dis-

appear. To get back to the normal word processor functions, press <EXIT>.

Options

1. Draft quality/high quality

If you only want to make a quick copy to check the content and layout, then draft quality is much quicker than high quality.

2. Single sheet/continuous

If you use continuous stationery then you will need to change this setting, or else the Amstrad will stop at the end of the first page and wait for you to change the paper. Also, the page length will probably be wrong (see 3 below).

3. Form length

This represents the maximum number of lines that can be printed on the sheet of paper. For continuous stationery it's the number of lines from the top of one sheet to the top of the next. As you change between single sheet and continuous you will see that the Amstrad has pre-set form lengths. These are convenient, but not obligatory. The default settings are 70 lines for single sheets and 66 for continuous stationery. However, if you use a different size than A4 single sheets then you will need to alter this. Above all, the form length set here should agree with the **page length** setting in the **file header** of the LocoScript document that you are printing out (see page 64). If you want to change the number of lines, place the cursor over the appropriate line, type in the new value and press <ENTER>.

4. Gap length

This setting shows the unprinted gap between pages of print. If (as with the default setting for continuous stationery) the page length is 66 lines, and the gap 5 lines, there will be 61 lines of print and 5 blank lines to make up one page. This is a good idea for printing out a BASIC program listing under CP/M, but can cause strange effects under LocoScript which prints in pages anyway. The safest thing to do is to change this setting to **zero**.

5. Paper out defeat

The printer has a safety device which detects when it is about to run out of paper. To save damage to the print head it stops printing about $\frac{3}{4}$ inch (2 cm) before the end. This can be very annoying if you want to print right to the bottom of the page when using single sheets. The paper out defeat allows you to do this.

When you first switch on the Amstrad it has these default settings for the printer parameters:

High quality
Single sheet paper
Paper length 70 lines
Gap length 5 lines

If you use continuous stationery, want draft quality print, or use a different size of paper, *you will need to change these settings every time that you switch the computer on*. Once set, they will remain so until you switch off, or leave LocoScript.

<f2>=Paper

This gives a pull-down menu that clears “Waiting for paper”. If this state occurs, it’s shown on the status line at the top of the screen. It’s possible for the Amstrad to believe that it’s out of paper when in fact it isn’t. This is probably because you are using continuous paper but haven’t changed the parameters under <f1>=Options (see above), in which case you should. It could be because you’ve put in a sheet of paper without using the paper feed mechanism. As this is the way that the printer tells when paper is loaded it thinks that there is none. Press <f2> and then <ENTER>.

<f3>=Actions

This pull-down menu performs four functions:

– Feed one line

If you press the <+> key, the paper in the printer will advance one line.

– Feed to top of form

If you move the menu cursor down using the <↓> key to cover

this line, pressing the <+> key will cause the printer to move the paper to the top of the next sheet. Where this is will depend on where the **top of form** is set (see below).

– *Set top of form*

This is fairly important when printing onto continuous stationery. When you load the paper, move it on (either with the paper loading knob, or with line feed described above) until the next row of perforations is just level with the top of the printer ribbon. Then, in the printer control state, press <f3>=Actions. Move the menu cursor down to **Set top of form** and press the <+> key. **Top of form** will be shown on the printer status line until the <EXIT> key is pressed to leave the printer control state.

– *Set left offset*

This allows you to set where the left hand edge of the text is printed, and is useful if you need to print the text in a special position on the paper. The value that you enter can be judged from the scale on the bail bar of the printer. The offset doesn't take the left margin of the document into account.

<f5>=Document/Reprint

If you interrupt printing part-way through a document, you can either continue printing from that point, or reprint from the beginning. Press <f5> and you will get a menu with these two options. Use the <↓> key to select the option you require. Note that this only works if you are in the middle of printing a document.

<f7>=Reset

This key is useful if you want to stop printing in the middle of a document because you see that something is wrong and you want to re-edit. If you press the <PTR> key, the printer will stop when it has finished the current line. If you press <EXIT> it will simply carry on where it left off, so instead press <f3> and feed to top of form. Then press <f7> reset, and abandon the printing with the <ENTER> key. If you abandon without first going to the top of the next form, the Amstrad will set the top of form to wherever the paper happened to be stopped.

<f8>=On/Off line

This menu allows you to switch the printer off-line so that if you arrest printing with the <PTR> key, it won't resume when you press the <EXIT> key. So if you need to make some adjustment involving the <EXIT> key, switch the printer off-line, make the adjustment and switch the printer on-line again when you are ready to resume printing.

PRINTING OUT DOCUMENTS

There are two ways of printing out a document. The first way occurs when exiting from editing text. On exit there is a menu which gives you four options:

- Finish editing**
- Save and continue**
- Save and print**
- Abandon edit**

Choosing the third option from this menu will make the Amstrad first save the text onto the disc, and then immediately print it out. But before choosing this option, make sure that the printer is set up to print out in the way that you require, because you aren't asked to check before the printing starts – see *Printer parameters* below.

The second way to print out some text is to use the **Print Document** mode from the disc management menu. Move the cursor to cover the file to be printed and press <P>. This sends text from a disc file directly to the printer. You are offered the option of printing out the entire document, or just one or more pages. These pages can be from any part of the document, and such things as headers, footers and page numbers are looked after. This can be very useful if you have just corrected one page within a long document (so long as the later pagination hasn't been affected), or need to repeat the last few pages. If you choose to print part of the document you are presented with a second pull-down menu which shows the number of pages that there are in the document, and lets you enter the page number on which to start printing, and the number of the last page that you want printed.

While the Amstrad is printing out, the screen will show the disc

management menu. You can now start creating or editing another document, provided that it is on the disc currently in the drive, or else on the memory disc. Both the printing and the editing will be slowed down slightly, particularly when the disc is being accessed for one or other of the tasks. However this is much more economic of time than waiting until the printing has been finished – particularly if the printing is in high-quality mode!

If you want to print out a document from one disc while editing one on another disc, then copy the file to be printed into the **memory disc** and print out from that. It's always much safer to edit directly from a disc, that way there is less chance of forgetting to save the file afterwards. Copy the file to be printed to the memory disc using <f3> in the normal way. Then change floppy discs so that the one holding the file to be edited is in the drive. Press <f1>=**Disc change**. Now check that the printer settings are correct, and with the cursor over the file to be printed in the memory disc press <P> and start printing out. When the Amstrad's screen returns to the disc management menu, you can move the cursor to the file to be edited and start editing in the normal way. Even in the middle of editing a document, you can go to the disc management menu to start printing out a document, provided that you don't have to change the disc with the file that you are editing.

TEMPORARY STOP

If you need to stop printing temporarily, for example to adjust the paper in the printer, this can be done by pressing the <PTR> key. The printer will stop, allowing you to make the adjustment. Pressing the <EXIT> key will allow the printer to resume printing exactly where it left off.

HOW TO ABANDON PRINTING

If you want to abandon printing, stop the printer by pressing the <PTR> key. This stops the printer at the end of the line that it is currently printing. If you are using continuous stationery, it's best to press <f3>=**Actions** and feed the paper to top of form. Then press <f7>=**Reset**. A pull-down menu will appear asking you to confirm that you want to abandon printing and reset. Press <ENTER> to

confirm, or press the <↓> and then <ENTER> if you want to change your mind.

DIRECT PRINTING

This mode, reached by pressing the <D> key from the disc management menu, allows you to use the Amstrad like an ordinary typewriter. This is mostly useful for filling in forms, etc where the paper needs to be lined up between each line of print. There are many advantages over a typewriter, for instance you can see and check each line before printing it out, and you can use all the Amstrad's range of character types and sizes.

When you press <D>, the disc management menu will clear and you will be presented with a single line. You can type to this in the same way as with creating or editing; the text can be corrected using the <←>, <→> and keys. You can continue to type beyond the end of the line, and the Amstrad will seem to accept as much as you care to enter. Printing out takes place as soon as you press <RETURN>. Although you can enter a lot of text, only the first 27 or so lines seem to get printed out. If you have used the cursor control keys to edit the text on the screen, it doesn't seem to matter where you place the <RETURN> symbol. Even if it's near the start of the text, all of it (up to 27 lines) will be printed out.

Once the text is printed out, it's lost from the Amstrad's memory, so it's not really advised for anything that you're likely to need to use again, more for quick notes to the milkman and that sort of thing. However if you *do* type something that you suddenly decide you'd like to keep, then before it's printed out, you can save it as a **block** in the normal way. Do resist the temptation to press the <RETURN> key before it's saved.

USING THE PRINTER UNDER CP/M

When you use the Amstrad under CP/M, the printer still has to be set for the type of paper that you're using. The default setting for the printer is single sheets of paper. Under CP/M, when you press the <PTR> button the printer control information is displayed across the **bottom** of the screen (see Figure 6.3). These are like the buttons on an ordinary printer. Use the <←> and <→> keys to move the

Printer:	On line	Top of form	LF	FF	Draft quality	P O defeat :On	Hex:Off	RESET
	Off line	at line ##			High quality	P O defeat :Off	Hex: On	

Figure 6.3 Printer Control Options under CP/M

printer control cursor from button to button and use the <+> and <-> keys to alter the settings.

Even if the printer controls are set to **on line**, the printer needs to be switched on and off from the keyboard using <ALT>+<P>. Pressing <ALT>+<P> for the first time switches the printer **on** (when the Amstrad will beep), pressing <ALT>+<P> again will switch the printer **off** (when the Amstrad will not beep).

Fortunately there is an easy way to prevent the printer from stopping at the bottom of each page when you are using continuous paper, and that's to use the **paper** utility under CP/M. After loading CP/M (and before doing anything else), and with the CP/M disc still in the drive, type:

paper 11 <RETURN>

This will set the printer for continuous stationery with a page length of 11 inches. You will find it most convenient to incorporate this instruction into a **submit file** as explained in Chapter 7, and as shown for the **PCW-INDEX** program in Chapter 8.

If you've forgotten to do this and the printer stops at the bottom of the first page, press the <PTR> key, use the <-> key to move the cursor to the **Waiting for paper** message and press <+>. Then press <EXIT> and printing will resume. You'll need to do this at the bottom of every page until you set the printer for continuous paper.

7 Using CP/M

INTRODUCTION

The microprocessor at the heart of the Amstrad PCW is a **Zilog Z80**. This processor has been around for some time; in fact it was introduced way back in 1977. The advantage of using it is that there is a great deal of commercial software available that will run on it. This is because there is a widely used operating system which allows the same software to be run on many different computers – so long as they used a Z80 as microprocessor. This operating system is known as **CP/M** (short for **control program for microcomputers**).

CP/M was developed by Digital Research Inc. The version of CP/M supplied with the Amstrad word processor is called **CP/M Plus**, which is synonymous with **CP/M 3.3**. Along with the CP/M come many utility programs, and indeed without these CP/M wouldn't be very useful. Unlike some other types of personal computers, CP/M machines don't store utilities in their internal memories. These are kept on disc and looked for when required. So, unless there is a disc holding the utilities in the drive, CP/M can't do very much. If you use your Amstrad solely for word processing with LocoScript, then you will only need to use CP/M for formatting discs. However, there is a vast number of CP/M programs available, many of which are available for use on the Amstrad word processor.

In CP/M micros, the disc drives are known by the drive letter. This is always followed by a colon (:). The standard Amstrad has only one physical drive, **drive a:**. If the second drive is fitted, this is **drive b:**. There is also the **memory disc**, drive **m:**. Most CP/M micros have two disc drives, one to hold the program disc (including CP/M

and the utilities), and one for the data. If your Amstrad only has one drive fitted, you can load the programs into the memory drive and use drive **a:** for the files. If the second drive is fitted, then use drive **a:** for the programs and drive **b:** for the files.

LOADING CP/M

CP/M is supplied on **side 2** of the software discs which came with the micro. The disc is *auto-booting*, that is to say, if the machine is switched on and the disc inserted into the disc drive it will load itself. If LocoScript is already loaded, you will need to press the **<SHIFT>**, **<EXTRA>** and **<EXIT>** keys. If still nothing happens, try pressing the **<SPACE>** bar.

Once loaded you will see the message:

**CP/M Plus Amstrad Consumer Electronics plc
v 1.4, 61 TPA, 1 disc drive, 112K drive m:**

This message will be slightly different if you have a PCW8512, a PCW8256 with more than one disc drive, or an optional Centronics/RS232 adaptor.

Underneath the message you will see:

A>

This is known as the **system prompt**. It tells you that CP/M is ready to do something, and that **drive a:** is the **selected** drive. You can change the selected drive simply by typing

m:<RETURN>

Now the prompt reads

M>

In CP/M, the drives are always referred to by their letter, which is always followed by a colon (:). It doesn't matter to CP/M whether you type in capital or lower case letters. It accepts either, but usually replies in capitals.

Even if you only have one disc drive fitted, you can still select drive **b:**, the Amstrad behaves as though it has two drives, instructing you to change the discs in the standard drive as necessary. If you have the second drive fitted, this becomes drive **b:**.

CP/M FILES

On the discs, information is stored in **files**. These can hold **languages** (CP/M, BASIC and LOGO are all supplied with the Amstrad as disc files), **utilities**, **programs** and **files of text or data**. Make sure that the CP/M disc is in the drive and that the prompt is **A>**. Then type:

dir<RETURN>

Now you will see all the files held on the disc. There can be up to 64 files on a single disc (256 in the second drive). This is the maximum number of entries in the disc directory, so even if there is room on the disc, no more files can be stored. Each file has a unique name (on one disc), known as the **file-name**. This can consist of up to eight letters, numbers or certain characters, followed by a full stop and a **file-name extension** of up to three more letters, etc. This is often used to show what sort of information is on the file. CP/M utility file-names usually end with **.COM**, basic programs with **.BAS**. If you type in a file-name, the Amstrad will look for that file in the selected drive. If you want to use a file in a drive other than the selected one, you must quote the **full file specification**. That might be:

m:pcwindex.bas

CP/M UTILITIES

CP/M comes with many utility programs – the ones with the file-name extension **.COM** in the directory. Limited versions of some of these are built into CP/M itself, but most have to be loaded from disc when required. Even the built-in ones will need to be loaded from disc if you want to use the **options** (see descriptions). In this chapter there are descriptions of the ones that you are most likely to use. Remember that *none* of these can be used when LocoScript is in the Amstrad's memory, although *some* of them can be used on LocoScript files from within CP/M. Be careful though, because LocoScript stores files differently to CP/M, so that files that are altered under CP/M could end up as unreadable. The utilities are called simply by typing their file-name followed by **<RETURN>**. CP/M will accept either capital or lower case letters, or even a mixture.

In the following examples, text within curly brackets { } is a *description* of what should be typed rather than what you *actually* type.

Square brackets [] are CP/M characters and so *are* typed, and anything like this – **<RETURN>** – refers to a dedicated key. If the utility is on disc rather than built-in, then it will only work if the disc holding the utility is in the drive currently selected. Hence if the utility is in the memory disc, the prompt should be **M>**, if not then type:

m:<RETURN>

to make it so. If you need a utility frequently, it's worth copying it (using **pip**) onto the disc holding the programs that you need to use the utility on.

If you're not used to entering CP/M instructions it's quite easy to make mistakes, and it's a terrible nuisance to have to re-type an instruction because you got one character wrong in the middle. Don't worry, you don't have to. If you make a mistake, and CP/M replies with a **?**, if you press the **<PASTE>** key in the top right hand corner of the keyboard, your last instruction will be entered again for you. Use the **<←>**, **<→>** and **<←DEL>** keys to delete the wrong characters and then insert the correct ones. When you have it correct, press **<RETURN>**.

Device

You'll need this utility if you want to use an alternative printer with the Amstrad. **Device** is on **side 3** of the discs supplied with your machine. The command for sending text to an alternative printer through the optional Centronics/RS232 adaptor (this only works under CP/M and *not* under LocoScript) is:

device lst:=cen

for a centronics printer, and:

device lst:=sio

for a serial printer.

The output baud rate is set by default to 9600 (for an explanation of *baud rate* see Chapter 10). This can be changed by adding the required baud rate after the command, eg:

device lst:=sio 4800

Dir (Built-in)

This prints out the contents of the disc in a *directory*. If you want the directory of the disc in the **currently selected drive**, then simply type:

dir<RETURN>

If you want the directory of a disc in a different drive, type:

dir{drive specification}<RETURN>

eg **dir m:**

There are several options for **dir**. If you are looking for a particular file, you could type:

dir a:pcwindex.bas<RETURN>

If the **pcwindex.bas** program *is* present in drive **a:**, then **dir** will confirm its presence by repeating the file-name. If not, the screen will show:

No file.

Wildcards can be used, *eg*:

dir m:*.bas

will list all the files containing BASIC programs on the file in drive **m:**. And **dir *.*** will list all the files on the currently selected drive. A further enhancement of this command is:

dir m:*.bas [drive=all]

This will list all the BASIC programs on any of the drives. There are many different permutations of options; a full list will be found in the CP/M manual supplied with the Amstrad, in section 5.1.

When you type **dir** you might see **SYSTEM FILE(S) EXIST**. To make the directory easier to read, it's possible to place some files in the **systems directory** (see **SET** page 95). These can be listed with the command:

dirsys<RETURN>

Disckit

The use of **disckit** is described in Chapter 3.

Erase (Built-in)

This utility erases files on the disc. To use it type:

era {file specification}<RETURN>

eg **era a:program.bas<RETURN>**

Caution. Use **erase** with great care; once a file has been erased under

CP/M it can't be recovered. The use of back-up copies is recommended (see **pip**). *Wildcards* can be used with erase, but this is rather dangerous, eg:

```
era *.*
```

will erase all the files on the currently selected drive (including the *systems files*), asking only *once* for confirmation. A slightly safer way is to use the extension:

```
era *.*[c]
```

This will at least ask for confirmation before deleting each file.

Paper

One of the things that new users find most confusing with the Amstrad is controlling the printer. This is usually because they don't realise that the printer expects **single sheets** of paper, unless set otherwise. This happens under CP/M just as with LocoScript. If you normally use continuous computer stationery in your printer, you can save yourself a lot of trouble by using the **paper** utility in a **submit** file when loading CP/M. The **paper** and **submit** utilities must be present on the disc, and the command in the submit file should be:

```
paper 11
```

This tells the printer to expect standard continuous computer stationery with 11-inch long pages, and to print at a line pitch of 6 lines per inch. If you want to set the printer for paper of another size, there is complete flexibility, but it needs a more complex command. See the *CP/M manual* supplied with the machine for details.

PIP

This stands for **peripheral interchange program**. It will copy a file from one place to another, usually from one disc to another, but it can even be used to copy a file to an external device. It's a very useful utility, but one that seems more complicated to use than it actually is. The basic command is:

```
pip {destination file specification} = {source file  
specification} <RETURN>
```

```
eg pip m:pcwindex.bas=a:pcwindex.bas<RETURN>
```

will copy **pcwindex.bas** from drive **a:** into drive **m:**.

This can be simplified in several ways. If you simply type:

```
pip<RETURN>
```

then you will be prompted for each entry. Otherwise you can eliminate unnecessary details,

eg **pip m:=a:pcwindex.bas<RETURN>**

This will copy the program **pcwindex.bas** presently on the disc in drive **a:** onto the memory disc. Wildcards work with **pip** too;

eg **pip m:=a:*. *<RETURN>**

will copy *all* the files in the normal directory on the disc in drive **a:** into the memory disc, but remember that the memory disc of the PCW8256 has a smaller capacity than drive **a:**. If you want to use this for transferring your utility programs to drive **m:**, you will need to prepare a disc with just the utilities that you need frequently. This is also a convenient way of making back-up copies of discs. Copy all the files into drive **m:**, and then change discs and turn the command round so:

pip a:=m:*. *<RETURN>

but remember that **pip** *must* be on one of the discs in the drives, and that the appropriate drive must be selected before using it.

If you want to copy files that are (or may be) in the systems directory, then **pip** will only find them if you use:

pip m:=a:mail232[r] or

pip a:=m:*. *[r]

This will cause **pip** to look in the systems directory as well. There must be no space immediately before **[r]**.

Pip can also be used to copy directly between two micro-floppy discs, even if only *one* physical disc drive is fitted. The Amstrad allows you to change the designation of the standard disc drive between **drive a:** and **drive b:**. In this way you can copy between **drive a:** and **drive b:** by changing over the discs in the drive. You will be prompted by the screen when to change the disc over in the drive.

Set

This utility will change the file type so that you can go some way to protecting it from accidental erasure, alteration or corruption. You can also make a file a **system** file, which means that it won't be listed by a simple **dir** command. This can make the directory easier to read when looking for files;

eg **set a:dir.com [sys]<RETURN>**

will make the copy of **dir.com** in drive **a:** a system file. Other options are:

- [dir]** converts a system file into a non-system file
- [ro]** makes a file so that it can only be *read* and not *written* to
- [rw]** converts a *read only* file to a *read/write* one (that is, one that can also be written to)

Set can also be extended to make a whole disc drive read only,
eg **set a:[ro]<RETURN>**

Note that if you want to copy *system files* using **pip**, you will need to use the option **[r]** after the pip command to show that the file is in the system directory which isn't usually searched.

Setkeys

All the keys on the Amstrad can be re-programmed to print any character when pressed. Some keys can even be programmed to print out a string of characters. This can be very useful in some cases, and an example is given in Chapter 11. To change the output from a key, the utility **setkeys** is used to load a file into the Amstrad containing the new definition(s). For example, supposing that you use the copyright sign © often. This can be found, but you would prefer it to be on the **<COPY>** key which isn't used in CP/M. You can re-program this key to print out the © when it's pressed. First you must create the file containing the new definition. This is done using the editing program **rpedit** which works under **BASIC**. Put your working copy of the CP/M disc into the Amstrad, let CP/M load up and then enter:

basic rpedit<RETURN>

This will load up **BASIC**, the BASIC program **rpedit**, and then run **rpedit**. **Rpedit** is rather like a very simple word processor which produces a file on the disc with no header. When **rpedit** is running you are faced with a menu page. Press **<f3>** to choose **edit a new page**. Then in the panel, enter the file-name **program.key**. The screen will clear to give an editing page. You can now type in the instructions to be contained in the file **program.key**.

All the keys on the Amstrad's keyboard have a **key number** (see page 109 of the *CP/M manual* supplied with the machine). The **<COPY>** key is key number 11. All the keys can be used in combi-

nation with one of the **shift keys**. These are <SHIFT>, <ALT> and <EXTRA>. The combination of shift key used with one of the other keys is known as the **shift state**. The required *shift state* is indicated with one or more letters,

eg S <SHIFT> + key
 A <ALT> + key
 E <EXT> + key
 SA <SHIFT> + <ALT> + key
 N key only.

The instruction to reprogram a key takes the form:
 {key number} {shift state} "{character}"

To change the <COPY> key to print out the *copyright sign* (©), the instruction is:

11 N "©"

You could also use

11 N "164"

because 164 is the **decimal value** of the © symbol.

This is just one key change. You could alter as many keys as you wish, but it's best to avoid re-programming any that are already used, as this would make using the keyboard rather difficult. To store the file on the disc, press <EXIT>.

After a short wait, you will find yourself back in the **rp**ed menu page. Press <f3> again to create another file. Call this one **profile.sub**. The contents should be:

setkeys program.key.

Now press <EXIT>, and then in the menu page press <EXIT> again. You will see the prompt A> to tell you that you're back in CP/M. Now you are ready to try your file. Re-boot CP/M by pressing <SHIFT>, <EXTRA> and <EXIT>. You will see the screen go light and the lines appear to show that CP/M is loading. Then you will see **setkeys program.key** to show that the contents of **profile.sub** has loaded. Once in CP/M, press <COPY> and the © symbol will be printed.

As well as programming a key with a single character, some keys on the Amstrad can be programmed with whole strings of characters. There is an example of how to do this in **Chapter 11 – Using MAIL232**.

Submit

Submit allows you to place a file on the disc so that the Amstrad automatically reads it as soon as CP/M is loaded, and obeys any commands contained there. This can be very useful for setting the computer up ready to use simply by placing the appropriate disc in the drive. Examples of this are given in **Chapter 8 – Using BASIC**, and **Chapter 11 – Using MAIL 232**.

To use submit the **submit.com** file must be present on the selected drive. When CP/M is loaded, the Amstrad automatically looks for a file called **profile.sub**, which it reads and obeys. The easiest way to create **profile.sub** is by using the **rped** program under BASIC. This is described under **SETKEYS** above.

Submit.com also allows you to have several submit files on the same disc, each to set up a different function. The files should all be saved with the file-name extension **.sub**, eg **pcwindex.sub**. Each could set up the Amstrad to run a different program, or set it up for different paper types, or what you will. Each **.sub** file would have to contain the appropriate commands, and would be called by typing:

```
submit pcwindex
```

Type (Built-in)

This lists the contents of an ASCII file (see page 167) to the Amstrad's screen. Type:

```
type {file specification}<RETURN>
```

The file will be listed to the screen a screenful at a time. To see the next screenful, press **<SPACE>**. If you want to list the file to the printer in order to get a hard copy, first check the printer parameters by pressing the **<PTR>** key and making any necessary alterations with the **<←>**, **<→>**, **<+>** and **<->** keys. Then get out of **<PTR>** with the **<EXIT>** key. You can switch the printer on using **Control-P** (**<ALT>+P**), then when you use **type**, anything sent to the screen is also printed out. Switch off the printer afterwards using **<ALT>+P** again. If you want to print out a long file without the bother of pressing **<SPACE>** at the end of each screenful, use:

```
type {file specification} [no page]<RETURN>
```

Make sure that you are using continuous paper (and the printer set to **paper 11**) if the file is longer than an A4 sheet.

8 BASIC Programs

INTRODUCTION TO MALLARD BASIC

The Basic language supplied with the Amstrad word processor is **Mallard BASIC** written by *Locomotive Software Ltd.* It's similar to the BASIC supplied with other Amstrad computers except that it doesn't support high-resolution graphics, colour or sound commands. It's very similar to the Microsoft BASIC supplied with many business machines, and this is appropriate, considering the uses for which the Amstrad word processor is intended. The BASIC manual supplied with the Amstrad contains a great deal of information for those who are already familiar with using BASIC, although some of it may be a little difficult to track down. Even though there are no high-resolution graphics, there are some low-resolution graphics characters, and Figure 8.1 shows these. They are used in the keyboard familiarisation program later in this chapter.

128	≡	136	≡	144	·	152	·
129	±	137	±	145	!	153	!
130		138	=	146	·	154	·
131	≡	139	≡	147	·	155	·
132	≡	140	≡	148	!	156	!
133		141	≡	149	!	157	!
134	≡	142	≡	150	!	158	!
135	≡	143	≡	151	!	159	!

Figure 8.1 Low-Resolution Graphics Characters

Mallard BASIC is never likely to become a *fashionable* computer language. It doesn't support **procedures**, so the programs have to use **GOTO** and **GOSUB**. The file handling is generally quite good, although some features like **length of file** are imprecise. Structured features that are supported are **WHILE** and **WEND**, and **user defined functions**. In order to *run* the BASIC, you need first to be in CP/M. Then with the CP/M disc still in the drive, type:

BASIC <RETURN> or
basic <RETURN>

Mallard BASIC accepts lower as well as upper case characters, even for BASIC reserved words. These are converted to upper case when you next list. Hence it's quite convenient to enter programs entirely in lower case, so that when listed, the BASIC reserved words are easily distinguished from all variable names.

BASIC will now run and show that it is ready with the prompt **Ok**. If you have a BASIC program on disc, the loading command is:

load "program name"<RETURN>

The second inverted commas may be omitted. Then type

run<RETURN>

and the program will start. If the program is on the same disc as BASIC, the two can be loaded together and auto-run using the single command:

basic program name

In this case no inverted commas are necessary, and this feature can be used to make auto-run discs by using this command in a **submit** file (see later).

When a Mallard BASIC program is listed, the scrolling can be paused, if required, by pressing <f5/f6>. The scrolling will start again if any character key (including the <SPACE> bar) is pressed. Another useful thing to know is that pressing <←> causes the last line entered to be repeated on the screen. If the line was entered incorrectly it can now be edited using the <←>, <→>, and <DEL→> <←DEL> keys.

BASIC PROGRAMS

The programs which follow in this chapter will show some of the ways that Mallard BASIC can be used.

(text continues on page 112)


```

10 REM =====
20 REM                      PCW-INDEX
30 REM          A database program for the Amstrad PCW
40 REM =====
50 OPTION RUN
60 REM ===== Init =====
70 esc$=CHR$(27)
80 home$=esc$+"H"
90 cls$=esc$+"E"+home$
100 revon$=esc$+"p"
110 revof$=esc$+"q"
120 curon$=esc$+"e"
130 curof$=esc$+"f"
140 clear$=esc$+"J"
150 end$=revof$+curon$
160 :
170 WIDTH 255
180 :
190 DIM field$(9),title$(9),len%(9),rec.field$(9),new,len%(9),
new,title$(9),sort$(9),cat$(10)
200 file$=""
210 records%=0
220 record%=0
230 fields%=6
240 total%=0
250 open%=0
260 header,file%=1
270 record,file%=2
280 :
290 DEF Fntab$(x%,y%)=esc$+"Y"+CHR$(32+y%)+CHR$(32+x%)
300 :
310 PRINT cls$
320 PRINT Fntab$(36,7);"=====
330 PRINT Fntab$(37,8);"PCW - I N D E X"
340 PRINT Fntab$(36,9);"=====
350 PRINT Fntab$(36,12);"by Stephen Milan"
360 PRINT Fntab$(5,20);"Which drive to use? ";curon$;
370 drive%=UPPER$(INKEY$);IF drive$="" THEN 370
380 IF INSTR("ABM",drive$)=0 THEN 370

```

```

390 OPTION FILES drive$
400 dummy%=FIND$( "*", "*" )
410 ;
420 REM ----- Menu -----
430 ;
440 PRINT cls$
450 PRINT home$;revon$;TAB(5);"File ";drive$;";";file$;TAB(50);"Number
of records";records$;TAB(93);revof$
460 PRINT Fntab$(10,3);"1) Load file from disc,"
470 PRINT Fntab$(10,6);"2) Add a record,"
480 PRINT Fntab$(10,9);"3) View records,"
490 PRINT Fntab$(10,12);"4) Sort records,"
500 PRINT Fntab$(10,15);"5) Print records,"
510 PRINT Fntab$(10,18);"6) Rename category,"
520 PRINT Fntab$(10,21);"7) Save file to disc,"
530 PRINT Fntab$(10,24);"8) Set-up new file,"
540 PRINT Fntab$(10,27);"EXIT to quit,"
550 PRINT Fntab$(10,30);"Enter choice ";curon$;
560 key%=INKEY$;IF key%="" THEN 560
570 IF key%=CHR$(27) AND open%=1 THEN GOSUB 990
580 IF key%=CHR$(27) THEN CLOSE;PRINT cls$;"Enter 'RUN <RETURN>' to
restart";PRINT curon$;END
590 key%=VAL(key%)
600 IF key%=0 OR key%>8 THEN 560
610 ON key% GOSUB 660,1130,1500,1890,2280,3150,990,3310
620 GOTO 440
630 END
640 ;
650 REM ----- Load new file -----
660 PRINT Fntab$(0,1);curof$;clear$
670 IF open%=1 THEN error$="File already loaded";GOSUB 3860;RETURN
680 GOSUB 910
690 PRINT Fntab$(5,10);"File name: ";curon$;
700 INPUT "",file$
710 file%=UPPER$(file$)
720 PRINT Fntab$(16,10);file$;curof$
730 IF FIND$(file$+".DAT")="" THEN error$="File doesn't exist";GOSUB
3860;RETURN
740 OPEN "I",header,file%,file$+".DAT"
750 INPUT # header,file%,records%,fields%

```

```
760 total%=0
770 FOR field%=1 TO fields%
780 INPUT # header,file%,title$(field%),len%(field%)
790 total%=total%+len%(field%)
800 NEXT field%
810 FOR cat%=1 TO 10
820 INPUT # header,file%,cat$(cat%)
830 NEXT cat%
840 CLOSE header,file%
850 PRINT FNTab$(4,15);records%;"records,"
860 GOSUB 3980
870 open%=1
880 RETURN
890 :
900 REM -- Catalogue disc --
910 found%=0
920 found$=FIND$("*",dat",found%+1)
930 IF found$="" THEN RETURN
940 found%=found%+1
950 PRINT FNTab$(70,5+found%*2);LEFT$(found$,8)
960 GOTO 920
970 :
980 REM ----- Save header file -----
990 PRINT FNTab$(0,1);clear$;curof$;
1000 IF open%=0 THEN error$="No file loaded";GOSUB 3860;RETURN
1010 OPEN "O",header,file%,file$+".DAT"
1020 WRITE # header,file%,records%,fields%
1030 FOR field%=1 TO fields%
1040 WRITE # header,file%,title$(field%),len%(field%)
1050 NEXT field%
1060 FOR cat%=1 TO 10
1070 WRITE # header,file%,cat$(cat%)
1080 NEXT cat%
1090 CLOSE header,file%
1100 RETURN
1110 :
1120 REM ----- Add Record -----
1130 records%=records%+1;record%=records%
1140 FOR field%=1 TO fields%
1150 field$(field%)=""
```

```

1160 NEXT field%
1170 ;
1180 REM ----- Edit Records -----
1190 PRINT FNTab$(0,1);curof$;clear$
1200 IF open%=0 THEN error$="No file loaded";GOSUB 3860;RETURN
1210 PRINT FNTab$(2,4);"Record ";record%
1220 FOR field%=1 TO fields%
1230 PRINT FNTab$(2,field%*2+5);title$(field%);FNTAB$(20,field%*2+5);
field$(field%);STRING$(len%(field%)-LEN(field$(field%)),"_")
1240 NEXT field%
1250 GOSUB 1440
1260 PRINT curon$
1270 FOR field%=1 TO fields%
1280 PRINT FNTab$(20,field%*2+5);;LINE INPUT "",line$
1290 IF line$<>"" THEN field$(field%)=line$
1300 IF LEN(field$(field%))>len%(field%) THEN PRINT FNTab$(20,field%*
2+5); STRING$(len%(field%),"_");SPACE$(69-len%(field%));GOTO 1280
1310 NEXT field%
1320 PRINT FNTab$(0,30);"1) File this record,      2) Cancel this
record,      3) Amend this record,      4) Menu,  ";
1330 key%=INKEY$;IF INSTR("1234"+CHR$(13),key%)=0 OR key%="" THEN 1330
1340 IF key%="4" THEN records%=records%-1;RETURN
1350 IF key%="2" THEN 1140
1360 IF key%="3" THEN 1190
1370 FOR field%=1 TO 9
1380 LSET rec,field$(field%)=field$(field%)
1390 NEXT field%
1400 PUT record,file%,record%
1410 RETURN
1420 ;
1430 REM ----- Print categories -----
1440 PRINT FNTab$(0,26);"Categories:";FNTab$(46,26);"A:";cat$(1);
FNTab$(68,26);"B:";cat$(2)
1450 PRINT FNTab$(2,27);"C:";cat$(3);FNTab$(24,27);"D:";cat$(4);
FNTab$(46,27);"E:";cat$(5);FNTab$(68,27);"F:";cat$(6)
1460 PRINT FNTab$(2,28);"G:";cat$(7);FNTab$(24,28);"H:";cat$(8);
FNTab$(46,28);"I:";cat$(9);FNTab$(68,28);"J:";cat$(10)
1470 RETURN
1480 ;

```



```

1490 REM ----- View Records -----
1500 PRINT Fntab$(0,1);curof$;clear$
1510 IF open%=0 THEN error$="No file loaded";GOSUB 3860;RETURN
1520 IF records%=0 THEN error$="No records to view";GOSUB 3860;RETURN
1530 record%=1
1540 IF records%=0 THEN RETURN
1550 GET record,file%,record%
1560 PRINT Fntab$(2,4);"Record: ";record%;" "
1570 FOR field%=1 TO fields%
1580 field$(field%)=rec,field$(field%)
1590 PRINT Fntab$(2,field%*2+5);title$(field%);Fntab$(20,field%*2+5);
field$(field%)
1600 NEXT field%
1610 GOSUB 1440
1620 PRINT Fntab$(0,30);"1) Next record,      2) Previous record,
3) Edit,      4) Delete,      5) Menu, ";curof$;
1630 key%=INKEY$;IF INSTR("12345"+CHR$(13),key%)=0 OR key%="" THEN 1630
1640 PRINT curof$;
1650 IF key%="1" OR key%=CHR$(13) AND record%<records% THEN
record%=record%+1;GOTO 1550
1660 IF key%="2" AND record%>1 THEN record%=record%-1;GOTO 1550
1670 IF key%="3" THEN GOSUB 1190;GOTO 1620
1680 IF key%="4" THEN GOSUB 1720;GOTO 1540
1690 RETURN
1700 ;
1710 REM ----- Delete record -----
1720 PRINT Fntab$(0,30);SPACE$(89);Fntab$(0,30);"Are you sure? (Y/N)
";curof$;
1730 key%=INKEY$
1740 IF UPPER$(key%)="N" THEN PRINT curof$;;RETURN
1750 IF UPPER$(key%)<>"Y" THEN 1730
1760 PRINT Fntab$(0,30);"Please wait --";SPACE$(75);
1770 FOR delete%=record%+1 TO records%
1780 PRINT Fntab$(15,30);records%-delete%;" ";
1790 GET record,file%,delete%
1800 PUT record,file%,delete%-1
1810 NEXT delete%
1820 records%=records%-1
1830 PRINT cls$

```

```

1840 PRINT home$;revon$;TAB(5);"File ; ";file$;TAB(50);"Number of
records";records$;TAB(93);revof$
1850 IF record%>records% THEN record%=records%
1860 RETURN
1870 ;
1880 REM ----- Sort -----
1890 PRINT FNTab$(0,1);clear$
1900 IF open%=0 THEN error$="No file loaded";GOSUB 3860;RETURN
1910 IF records%<2 THEN error$="Not enough records to sort";GOSUB
3860;RETURN
1920 PRINT FNTab$(2,4);"Sort by which field:"
1930 FOR field%=1 TO fields%
1940 PRINT FNTab$(9,5+field%*2);STR$(field%);" " ;title$(field%)
1950 NEXT field%
1960 PRINT FNTab$(10,8+fields%*2);"EXIT to return to menu"
1970 PRINT FNTab$(10,11+fields%*2);"Enter choice ";curon$;
1980 key$=INKEY$;IF key$="" THEN 1980
1990 IF key$=esc$ THEN RETURN
2000 IF VAL(key$)=0 OR VAL(key$)>fields% THEN 1980
2010 sort%=VAL(key$)
2020 PRINT sort$;FNTab$(50,15);"Please wait -- ";curof$
2030 ;
2040 FOR sort1%=records% TO 2 STEP -1
2050 PRINT FNTab$(64,15);sort1%-1;" "
2060 FOR sort2%=2 TO sort1%
2070 GET record,file$,sort2%
2080 sort$=rec.field$(sort%)
2090 GET record,file$,(sort2%-1)
2100 IF sort$>rec.field$(sort%) THEN 2240
2110 ;
2120 REM ----- swap record sort2% and sort2%-1 -----
2130 FOR field%=1 TO 9
2140 sort$(field%)=rec.field$(field%)
2150 NEXT field%
2160 GET record,file$,sort2%
2170 PUT record,file$,(sort2%-1)
2180 FOR field%=1 TO 9
2190 LSET rec.field$(field%)=sort$(field%)
2200 NEXT field%
2210 PUT record,file$,sort2%

```

```
2220 :
2230 REM -- rest of loop --
2240 NEXT sort2%,sort1%
2250 RETURN
2260 :
2270 REM ----- Print -----
2280 PRINT Fntab$(0,1);clear$
2290 IF open%=0 THEN error$="No file loaded";GOSUB 3860:RETURN
2300 PRINT Fntab$(2,5);clear$;"Print to screen, printer or label?  ";
2310 LINE INPUT "", to$
2320 to$=UPPER$(to$)
2330 IF LEFT$(to$,1)<>"S" AND LEFT$(to$,1)<>"P" AND LEFT$(to$,1)<>"L"
THEN 2300
2340 IF LEFT$(to$,1)="L" THEN GOSUB 2760
2350 PRINT Fntab$(0,1);clear$;FNTAB$(0,2);"Enter characters or strings
to be searched for,"
2360 :
2370 FOR field%=1 TO fields%
2380 PRINT
Fntab$(2,3+field%*2);title$(field%);Fntab$(20,3+field%*2);STRING$(len$(
field%),"_")
2390 NEXT field%
2400 GOSUB 1440
2410 FOR field%=1 TO fields%
2420 PRINT Fntab$(20,3+field%*2);
2430 LINE INPUT "",sort$(field%)
2440 IF LEN(sort$(field%))>len$(field%) THEN PRINT
Fntab$(20,3+field%*2);STRING$(len$(field%),"_");GOTO 2420
2450 NEXT field%
2460 :
2470 PRINT Fntab$(0,1);clear$
2480 FOR record%=1 TO records%
2490 GET record,file%,record%
2500 ok%=1
2510 FOR cat%=1 TO LEN(sort$(1))
2520 IF INSTR(UPPER$(rec,field$(1)),UPPER$(MID$(sort$(1),cat%,1)))=0
THEN ok%=0
2530 NEXT cat%
2540 IF ok%=0 THEN GOTO 2640
2550 FOR field%=2 TO fields%
```

```

2560 IF INSTR(UPPER$(rec,field$(field%)),UPPER$(sort$(field%)))=0 THEN
ok%=0
2570 NEXT field%
2580 IF ok%=0 THEN 2640
2590 IF LEFT$(to$,1)="L" THEN GOSUB 2960;GOTO 2640
2600 FOR field%=2 TO fields%
2610 IF LEFT$(to$,1)="S" THEN PRINT rec,field$(field%) ELSE LPRINT
rec,field$(field%)
2620 NEXT field%
2630 IF LEFT$(to$,1)="S" THEN PRINT ELSE LPRINT
2640 NEXT record%
2650 LPRINT CHR$(27);CHR$(70);
2660 IF LEFT$(to$,1)="P" OR LEFT$(to$,1)="L" THEN RETURN
2670 PRINT;PRINT"1) Print          2) Return to menu. ";
2680 key%=INKEY$
2690 IF key%="" THEN 2680
2700 IF key%="2" THEN RETURN
2710 IF key%<>"1" THEN 2680
2720 to$="P"
2730 GOTO 2470
2740 ;
2750 REM ----- input label format -----
2760 PRINT FNTab$(0,1);clear$
2770 PRINT FNTab$(0,5);"Input the number of lines per label (8 usual, 9
max).";
2780 PRINT FNTab$(0,7);"Number of lines ";
2790 INPUT "",lines%
2800 IF lines%>9 THEN 2760
2810 PRINT FNTab$(0,1);clear$
2820 FOR field% = 2 TO fields%
2830 PRINT FNTab$(2,1+field%);field%-1;"      ";titles$(field%)
2840 NEXT field%
2850 FOR line% = 1 TO lines%
2860 PRINT FNTab$(2,line%*2+10);"Line";line%;"      ";STRING$(20,"_")
2870 NEXT line%
2880 FOR line% = 1 TO lines%
2890 PRINT FNTab$(14,line%*2+10);
2900 LINE INPUT "", format$(line%)
2910 NEXT line%
2920 LPRINT CHR$(27);CHR$(69);

```



```

2930 RETURN
2940 ;
2950 REM ----- print label -----
2960 FOR line% = 1 TO lines%
2970 IF LEN(format$(line%)) = 0 THEN LPRINT;GOTO 3100
2980 FOR pos% = 1 TO LEN(format$(line%))
2990 char% = MID$(format$(line%),pos%,1)
3000 IF INSTR(LEFT$("123456789",fields%),char%) = 0 THEN 3070
3010 field% = rec,field%(VAL(char%)+1)
3020 IF RIGHT$(field%,1) <> " " THEN 3050
3030 field% = LEFT$(field%,LEN(field%)-1)
3040 IF LEN(field%)>0 THEN 3020
3050 LPRINT field%;
3060 GOTO 3080
3070 LPRINT char%;
3080 NEXT pos%
3090 LPRINT
3100 NEXT line%
3110 LPRINT
3120 RETURN
3130 ;
3140 REM ----- Rename Category -----
3150 PRINT Fntab$(0,1);clear$
3160 IF open%=0 THEN error$="No file loaded";GOSUB 3860;RETURN
3170 FOR cat%=1 TO 10
3180 PRINT Fntab$(7,cat%*2+5);CHR$(64+cat%);"; ";cat$(cat%);STRING$(18-LEN(cat$(cat%)),"_")
3190 NEXT cat%
3200 FOR cat%=1 TO 10
3210 PRINT Fntab$(11,cat%*2+5);
3220 LINE INPUT "", text$
3230 IF text$="" THEN 3260
3240 IF LEN(text$)>18 THEN 3210
3250 cat$(cat%)=text$
3260 NEXT cat%
3270 RETURN
3280 ;
3290 ;
3300 REM ----- Set up -----
3310 PRINT Fntab$(0,1);clear$

```

```

3320 IF open%=1 THEN error$="File already loaded";GOSUB 3860;RETURN
3330 PRINT Fntab$(0,6);"File name          ";
3340 PRINT Fntab$(0,8);"Number of fields (1-8) ";
3350 PRINT Fntab$(25,6);;LINE INPUT "",text$
3360 IF text$="" OR LEN(text$)>8 THEN error$="Invalid file name";GOSUB
3860;GOTO 3310
3370 new.file$=UPPER$(text$);PRINT Fntab$(25,6);new.file$
3380 IF FIND$(text$+ ".DAT",1)>" " THEN error$="File already
exists";GOSUB 3860;GOTO 3310
3390 ;
3400 PRINT Fntab$(25,8);SPACE$(60);Fntab$(25,8);;LINE INPUT "",text$
3410 IF VAL(text$)<1 OR VAL(text$)>8 THEN 3400
3420 new.fields%=VAL(text$)+1
3430 PRINT Fntab$(11,11);"Field title";SPACE$(30);"Field length"
3440 FOR field%=2 TO new.fields%
3450 PRINT Fntab$(0,9+field%*2);"Field";field%-1;"; "
3460 NEXT field%
3470 new.title$(1)="Categories";new.len%(1)=10
3480 total%=10
3490 FOR field%=2 TO new.fields%
3500 PRINT Fntab$(11,9+field%*2);STRING$(15,"_");Fntab$(11,9+field%*2);
;LINE INPUT "",text$
3510 IF text$="" OR LEN(text$)>15 THEN 3500
3520 new.title$(field%)=text$
3530 PRINT Fntab$(52,9+field%*2);STRING$(2,"_");Fntab$(52,9+field%*2);
;LINE INPUT "",text$
3540 IF VAL(text$)=0 OR VAL(text$)>65 THEN 3530
3550 new.len%(field%)=VAL(text$)
3560 total%=total%+VAL(text$)
3570 NEXT field%
3580 ;
3590 PRINT Fntab$(70,15);"1) Accept"
3600 PRINT Fntab$(70,17);"2) Re-try"
3610 PRINT Fntab$(70,19);"3) Exit"
3620 key$=INKEY$;IF key$="" THEN 3620
3630 IF key$="2" THEN 3310
3640 IF key$="3" THEN PRINT Fntab$(0,3);clear$;RETURN
3650 IF key$("<") THEN 3620
3660 ;
3670 file$=new.file$

```

```

3680 fields%=new.fields%
3690 records%=0;record%=0
3700 FOR field%=1 TO fields%
3710 title$(field%)=new,title$(field%)
3720 new,title$(field%)=""
3730 len%(field%)=new,len%(field%)
3740 NEXT field%
3750 IF fields%<9 THEN FOR field%=fields%+1 TO 9:len%(field%)=0:NEXT
field%
3760 FOR cat%=1 TO 10
3770 cat$(cat%)=""
3780 NEXT cat%
3790 PRINT Fntab$(0,3);clear$
3800 ;
3810 open%=1;GOSUB 990;GOSUB 3980
3820 ;
3830 RETURN
3840 ;
3850 REM ----- Display Errors -----
3860 space%=INT((29-LEN(error%))/2)
3870 PRINT Fntab$(30,4);revon%;CHR$(134);STRING$(29,138);CHR$(140)
3880 PRINT Fntab$(30,5);CHR$(133);"          ERROR          ";
CHR$(133)
3890 PRINT Fntab$(30,6);CHR$(133);SPC(space%);error%;SPC(29-space%-
LEN(error%));CHR$(133)
3900 PRINT Fntab$(30,7);CHR$(135);STRING$(29,138);CHR$(141)
3910 PRINT Fntab$(30,8);CHR$(133);"    Press SPACE to continue
";CHR$(133)
3920 PRINT Fntab$(30,9);CHR$(131);STRING$(29,138);CHR$(137);revof$
3930 key%=INKEY$
3940 IF key$<>" " THEN GOTO 3930
3950 RETURN
3960 ;
3970 REM ----- Open record file -----
3980 MEMORY ,,,total%
3990 OPEN "R",record,file%,file%+".REC",total%
4000 FIELD record,file%,len%(1) AS rec.field$(1),len%(2) AS
rec.field$(2),len%(3) AS rec.field$(3),len%(4) AS rec.field$(4),len%(5)
AS rec.field$(5),len%(6) AS rec.field$(6),len%(7) AS
rec.field$(7),len%(8) AS rec.field$(8),len%(9) AS rec.field$(9)
4010 RETURN

```

(program ends)

PCW-INDEX

This program allows you to keep a mail list, or any other type of list in a disc file, or rather two disc files, as you will see. The program is rather like a card index. The file contains **records**, each equivalent to a card, and each record can contain a number of **fields**; that is, spaces for information (see Figure 8.2).

FILE

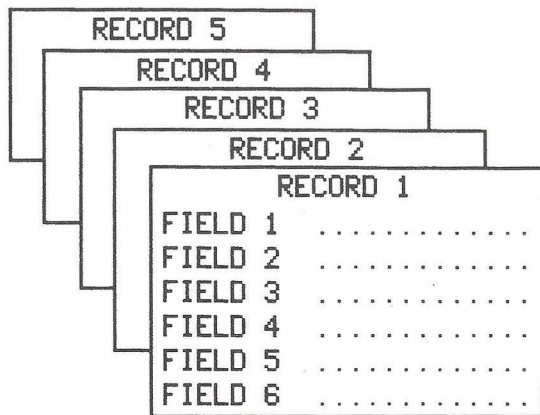


Figure 8.2 Card File

When the program is first loaded, you will be asked which drive you want to use. If you only have one drive then the answer will be A, although you *could* use drive M. Once the program is loaded you can change the disc in drive A for the one holding the data files. If you have a second disc drive then you can choose to use that for the data. Then you are presented with the main menu:

1. Load a file from disc
2. Add a record
3. View records
4. Sort records
5. Print records
6. Rename categories
7. Save file to disc
8. Set up new file

If you don't have any data files yet, then the first thing that you need to do is to create a new file. This is option 8. You must give the file a name, and define a layout for the records. Figure 8.3 is an example of a create file screen.

```

File #: _____ Number of records 0

File name      : BOOKLIST
Number of fields (1-8) : 6

Field title          Field length
Field 1 : Title_____ 50
Field 2 : Author_____ 25      1) Accept
Field 3 : Publisher_____ 20    2) Re-try
Field 4 : ISBN number___ 15     3) Exit
Field 5 : Year_____ 4_
Field 6 : Price_____ 5_

```

Figure 8.3 PCW-INDEX Create File Screen

Each record can contain up to 8 fields; each field can be a different size, up to a maximum of 65 characters each. This is to reserve the space on the disc. Of course any particular field can be defined to hold fewer characters, but once a size has been defined, each field will occupy that amount of space on the disc. It's best to spend some time deciding the layout of records for a new file. On the one hand, don't use up any more space than you think you'll need as this will reduce the ultimate number of records that can be kept, and probably slow the program down. On the other, and perhaps more importantly, it will be very annoying to have entered a lot of records only to find that the fields really aren't big enough and to have to go back and start again. It may be possible with a lot of jiggery pokery to read a .REC file with a new .DAT header with larger fields, but it will have been a lot easier to have got the record specification right from the start.

Once the record layout has been specified, the program opens two files on the disc under the file-name that you gave, one with the file-

name extension **.DAT** containing the names and sizes of the fields. The other has the file-name extension **.REC** which will contain all the actual records. Now you can enter data into them.

Option 2 is **add a record**. This brings up the next free record for you to enter data into the fields. You can add a record at any time, up to the limit of the space on the disc. Figure 8.4 shows a typical add a record screen.

File A:\BOOK\191		Number of records 23	
Record 24			
Categories	e_____		
Title	Using the Amstrad Word Processor_____		
Author	Michael Milan_____		
Publisher	MCC Publications____		
ISBN number	0 85012 567 7__		
Year	1986		
Price	8.50_		
Categories:			
C:Romance	D:Science fiction	A:General fiction	B:Thriller
G:Cookery	H:Gardening	E:Computer	F:Travel
		I:Do it yourself	J:Foreign language
1) File this record. 2) Cancel this record. 3) Amend this record. 4) Menu. █			

Figure 8.4 Add a Record Screen

Once the disc is full, if you try to add more records you will get an error message and the program will end. You may lose the last few records, but the rest will be safe. If this happens, clear some room on the disc, or copy the two files (**.DAT** and **.REC**) onto another disc with more room and carry on. **Don't** use a disc that you are using for LocoScript files.

Records can be added or deleted later. Each record can also be assigned to up to ten **categories**, which can be used later during searches. You can give the categories names using option 6 on the menu. You can view the records or print them out. Option 3, **View records**, allows you to look through the file. Option 5 allows you to

view or print out a selection of the records. If you choose to print out labels (although it could just as easily be on plain paper), you can manipulate the format. Numbers from 0-9 are taken to represent field numbers, and the fields are printed out where these occur. Other characters and spaces are printed out as entered. Self adhesive labels are available on a backing with holes down either side to fit the tractor feed of the printer. The most common size is $1\frac{7}{16}$ inches high, and $3\frac{1}{2}$ or 4 inches wide. These will hold eight lines of text. The program allows you to adjust the printout for other sizes. Figure 8.5 is an example of a label formatting screen, and Figure 8.6 shows a typical label.

File A:BOOKLIST		Number of records 24
1	Title	
2	Author	
3	Publisher	
4	ISBN number	
5	Year	
6	Price	
Line 1	Title_____	
Line 2	1_____	
Line 3	Author_____	
Line 4	2_____	
Line 5	Published by_____	
Line 6	3_____	
Line 7	5 4_____	
Line 8	£6_____	

Figure 8.5 Label Formatting Screen

```

Title
Using the Amstrad Word Processor
Author
Michael Milan
Published by
NCC Publications
1986 0 85012 567 7
£8.50

```

Figure 8.6 Label

Before you print out the records you can instruct the program to search for certain strings, for categories, or groups of categories. The screen shows all the fields and categories. If you just press <RETURN> in each field, then no search is made on that field. But if any letter or string is entered, then only records where that letter or string occurs in that field will be printed. The search doesn't distinguish between upper and lower case. When specifying categories, it doesn't matter what order the categories are entered, or whether or not they are adjacent on the records. If the file contains details of your musical record collection, then searching for **mon** could turn up records containing **Harmony**, **Montiverdi**, and **Lass of Richmond Hill**.

Option 4, **Sort records**, will sort the records into alphabetical order of any of the fields that you choose. Sorting takes some time, a file with 60 records takes about 6 minutes, longer files are likely to take disproportionately longer. There is quite a lot of disc activity for the first part of the sort at least.

When you have finished with the current file, press <EXIT> on the main menu. This will make sure that the files are closed properly. To load another file, type **run**<RETURN>.

Entering PCW-INDEX

This program may seem rather daunting to enter into the computer, but it works quite well and is very useful. You could expect to pay more than the cost of this book for anything similar. Entering it shouldn't be too difficult, provided that you check very carefully for typing errors. As Mallard BASIC relies on **GOTOs** and **GOSUBs** the line numbers are significant and must be entered as shown in the listing. The listing has been printed out to fill the page of this book, and not as it will appear across the Amstrad's screen. Remember not to press the <RETURN> key until before the next line number. Be careful to distinguish correctly between **field\$**, **field%** and **fields%**. Line 50, **OPTION RUN**, should be entered **last**, and only when you are sure that the program is working correctly. Its purpose is to disable the <STOP> key. This is dangerously close to the <1> key, and pressing it accidentally will cause you to break the program and lose a lot of data. With the <STOP> key disabled, the only way out of the program is to press the <EXIT> key, which saves all the data to

the disc safely, and closes the files. This is a computerised equivalent of putting all the cards back in the index and shutting the drawer!

Enter the program, saving it to disc *frequently* to avoid disaster. It's a good idea to save it on more than one disc at the same time. Files accidentally erased under CP/M cannot be recovered. Call the finished version **pcwindex**.

Once the program has been loaded into the computer, the disc can be removed from the drive and a data disc inserted. If you have the second drive, the data can be on a disc in that drive.

Creating a PCW-INDEX Disc

The whole program will be much more convenient to use if you make up a special PCW-INDEX disc. This should contain the following files:

J14CPM3.EMS (CP/M*)
BASIC.COM
SUBMIT.COM
PCWINDEX.BAS
PAPER.COM
PROFILE.SUB

The following are needed temporarily:

RPED.BAS
SET.COM

(*Your version may have a different number – check with **dir**)

To create the PCW-INDEX disc, take an empty, formatted disc and copy onto it all the above files except **PROFILE.SUB** (we will create this using **RPED**).

From CP/M type:

basic rped<RETURN>

This will *load* **basic** and **rped**, and then *run* **rped**. When the main **rped** menu comes up, press **<f1>** to edit a new file. Give it the name:
profile.sub

Note that **rped** is rather inconsistent here with other programs supplied with the Amstrad; in order to get the cursor to the second part of the file-name (the file-name extension), you must press either the **full stop <.>** or **<->**. Then press **<ENTER>** and the screen will

change to the editing screen. Now type in the required instructions for this file:

paper 11
basic pcwindex

When you have finished, press **<EXIT>** twice – once to save the new file to the disc, and the other to leave rped. Once this has been done, test the disc by pressing **<SHIFT>+<EXTRA>+<EXIT>**. The Amstrad should load CP/M, basic and pcwindex, and then *run* pcwindex. Once you are satisfied, use **SET** to make the required files **read-only** so that they cannot be accidentally erased:

set J14CPM3.EMS [ro] **<RETURN>**
set BASIC.COM [ro] **<RETURN>**
set SUBMIT.COM [ro] **<RETURN>**
set PCWINDEX.BAS [ro] **<RETURN>**
set PAPER.COM [ro] **<RETURN>**
set PROFILE.SUB [ro] **<RETURN>**

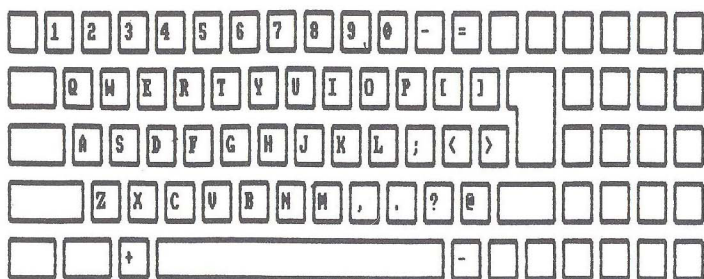
If you wish, you can also use **set** to make all these files **system** files so that they don't normally show up with **dir**. Then you can erase **RPED.BAS** and **SET.COM**. This will leave you with enough room on the disc for a small file, however you can always remove this disc once the program is loaded and use an empty (formatted) one in its place, or else use the second drive if fitted. An empty disc here would have enough room for an enormous file. You should keep **back-up** copies of your data. To do this, copy the whole data disc using **diskit**, or the data files using **pip**. Remember that there are two files (**.DAT** and **.REC**) for each set of data.

PERSONAL DATA

If you use your Amstrad to record personal details of people, other than just their names and addresses, then under the **Data Protection Act 1984** currently in force in the UK, you may be required to register with the **Data Protection Registrar**. Other countries may have similar regulations.

TYPING TUTOR

This program will help new users become familiar with the keyboard (although arguably they will become more familiar with the



- 1) Letter test with keyboard highlight.
- 2) Letter test without keyboard highlight.
- 3) Sentence test.
- 4) Load progress.
- 5) Save progress.

Average time: 0.694 secs.

Total percentage accuracy: 93

Figure 8.7 Keyboard Program Screen

keyboard by typing in the program than using it!). The screen will show a diagram of the keyboard (see Figure 8.7). The user is given the choice of a keyboard test with or without the keyboard highlighted, or a sentence test. The time taken to press the appropriate keys is recorded. At first the keys to be found are selected randomly. Later the keys with the poorest performance are selected preferentially so that the user gets more practice with these. Performance can be recorded to disc and reloaded later to continue.

The program is interesting in that it uses low-resolution graphics characters to create a diagram of the keyboard on the screen. The most critical line to enter is line 560. This must be entered exactly as follows:

```
560 keys$=" {2 spaces}1!2 {1 space}3£4$5%6'7&8*9(0)
-_-+ {15 spaces}qQwWeErRtTyYuUiIoOpP[{ ] {12 spaces}
aAsSdDfFgGhHjJkKlL;:$<#> {10 spaces}zZxXcCvVbB
nNmM,.,/'?1/2@"
```

More sentence tests can be added, they are held in **DATA statements** at the end of the program. The first DATA statement for each starts with the number of lines that it covers. The value of the variable **sentences%** in line 550 must be altered to show the number of sentences, irrespective of the number of lines in each.

```

10 REM *****
20 REM *
30 REM *      TYPING TUTOR      *
40 REM *
50 REM *    by Stephen  Milan   *
60 REM *
70 REM *    for the Amstrad PCW  *
80 REM *
90 REM *****
100 esc$=CHR$(27)
110 home$=esc$+"H"
120 cls$=esc$+"E"+home$
130 revon$=esc$+"p"
140 revof$=esc$+"q"
150 curon$=esc$+"e"
160 curof$=esc$+"f"
170 clear$=esc$+"J"
180 DEF FNtab$(x%,y%)=esc$+"V"+CHR$(32+y%)+CHR$(32+x%)
190 DEF FNrand(range%)=INT(RND(1)*range%)+1
200 ;
210 WIDTH 255
220 ;
230 REM ----- Main Loop -----
240 GOSUB 460
250 PRINT cls$;curof$
260 GOSUB 1940
270 PRINT FNtab$(0,16);clear$;
280 PRINT FNtab$(20,17);"1) Letter test with keyboard highlight,"
290 PRINT FNtab$(20,19);"2) Letter test without keyboard highlight,"
300 PRINT FNtab$(20,21);"3) Sentence test,"
310 PRINT FNtab$(20,23);"4) Load progress,"
320 PRINT FNtab$(20,25);"5) Save progress,"
330 PRINT FNtab$(10,30);"Average time:";INT(average*1000)
/1000;"secs.";FNtab$(48,30);"Total percentage accuracy:";
INT(accuracy*0,5);
340 random%=0
350 key$=INKEY$;random%=(random%+1 MOD 32000);IF key$="" THEN GOTO 350
360 IF INSTR("12345",key$)=0 THEN GOTO 350
370 PRINT FNtab$(0,17);clear$;

```



```

380 RANDOMIZE random%
390 IF key$="1" THEN keyboard%=1 ELSE keyboard%=0
400 IF key$="1" OR key$="2" THEN GOSUB 600:GOTO 270
410 IF key$="3" THEN GOSUB 1210:GOTO 270
420 IF key$="4" THEN GOSUB 1720:GOTO 270
430 IF key$="5" THEN GOSUB 1830:GOTO 270
440 :
450 REM ----- Initialise -----
460 DIM key$(83),len$(83),x$(83),y$(83),need$(83),letter$(83),line$(5)
470 FOR key%=1 TO 83
480 need$(key%)=1
490 READ letter$(key%)
500 NEXT key%
510 total%=83
520 average=0
530 accuracy=0
540 convert%=230
550 sentences%=8
560 keys$=" 112 3&4$5&6'7&8*9(0)-_+=+          qQwWeErRtTyYuUiIoD
pPl(1)          aAsSdDfFgGhHjJkKlL;:;<=>      zZxXcCvVbBnNmM,...
/?%@"
570 RETURN
580 :
590 REM -- Test keys --
600 PRINT FNtab$(0,18);"Press;"
610 :
620 right%=0
630 wrong%=0
640 FOR number%=1 TO 20
650 prob%=FNrand(total%)
660 which%=0
670 key%=0
680 key%=key%+1
690 which%=which%+need$(key%)
700 IF which%<prob% THEN GOTO 680
710 IF LEN(key$(key%))=0 THEN GOTO 650
720 :
730 shift%=FNrand(2)
740 IF key%=3 THEN shift%=1
750 PRINT FNtab$(43,18);MID$(key$(key%),shift%,1)

```

```

760 IF keyboard%=0 THEN GOTO 840
770 IF shift%=1 THEN GOTO 800
780 x%=x%(55)+6;y%=y%(55);len%=len%(55);letter$="      "
790 PRINT revon$;GOSUB 2050;PRINT revof$
800 x%=x%(key%)+6;y%=y%(key%);len%=len%(key%);letter%=letter$(key%)+ " "
810 IF key%=75 THEN letter$=SPACE$(len%(key%))
820 PRINT revon$;GOSUB 2050;PRINT revof$
830 ;
840 loop%=0
850 loop%=loop%+1
860 pressed$=INKEY$
870 IF pressed$="" THEN GOTO 850
880 time=loop%/convert%
890 ;
900 IF pressed$=MID$(key$(key%),shift%,1) THEN 980
910 IF need%(key%)>20 THEN GOTO 940
920 need%(key%)=need%(key%)+3
930 total%=total%+3
940 wrong%=wrong%+1
950 PRINT FNTab$(39,20);"INCORRECT"
960 GOTO 1030
970 ;
980 IF time>2 AND need%(key%)<20 THEN need%(key%)=need%(key%)+1;
total%=total%+1
990 IF time<1 AND need%(key%)>1 THEN need%(key%)=need%(key%)-1;
total%=total%-1
1000 right%=right%+1
1010 PRINT FNTab$(40,20);"CORRECT"
1020 ;
1030 IF average=0 THEN average=time ELSE average=(average+time)/2
1040 IF accuracy=0 THEN accuracy=(right%/number%)*100 ELSE accuracy=
(accuracy+INT((right%/number%)*100+0.5))/2
1050 PRINT FNTab$(0,22);"Average time per key ";INT(average*1000)
/1000;" "
1060 PRINT FNTab$(0,24);"Number correct ";right%
1070 PRINT FNTab$(0,26);"Number incorrect ";wrong%
1080 PRINT FNTab$(0,28);"Percentage correct ";INT((right%/number%)*
100+0.5)
1090 FOR delay%=0 TO 500:NEXT delay%
1100 GOSUB 2050

```

```

1110 x%=x%(55)+6;y%=y%(55);len%=len%(55);letter$=" "
1120 GOSUB 2050
1130 PRINT Fntab$(43,18);" "
1140 PRINT Fntab$(39,20);SPACE$(9)
1150 RANDOMIZE loop%
1160 FOR delay%=0 TO 500:NEXT delay%
1170 NEXT number%
1180 RETURN
1190 ;
1200 REM ----- Test sentences -----
1210 right%=0
1220 wrong%=0
1230 RESTORE 2490
1240 sentence%=FNrand(sentences%)
1250 IF sentence%=1 THEN GOTO 1270
1260 FOR find%=1 TO sentence%-1:READ lines%;FOR line%=1 TO lines%;READ
line$;NEXT line%,find%
1270 READ lines%
1280 FOR line%=1 TO lines%
1290 READ line$(line%)
1300 PRINT Fntab$(8,13+line%*3);line$(line%)
1310 NEXT line%
1320 ;
1330 PRINT curon$;Fntab$(60,30);"Average time:";
1340 FOR line%=1 TO lines%
1350 FOR letter%=1 TO LEN(line$(line%))
1360 PRINT Fntab$(7+letter%,14+line%*3);
1370 ;
1380 loop%=0
1390 loop%=loop%+1
1400 pressed$=INKEY$
1410 IF pressed$="" THEN GOTO 1390
1420 time=loop%/convert%
1430 ;
1440 IF average=0 THEN average=time ELSE average=(average+time)/2
1450 key$=MID$(line$(line%),letter%,1)
1460 IF key$=" " THEN key%=75 ELSE key%=INT(INSTR(keys$,key$)/2+0.5)
1470 IF pressed$=key$ AND need%(key%)>1 THEN need%(key%)=need%(key%)-1
1480 IF pressed$=key$ THEN right%=right%+1;GOTO 1520
1490 IF need%(key%)<20 THEN need%(key%)=need%(key%)+3

```

```

1500 wrong%=wrong%+1
1510 PRINT revon$;
1520 PRINT pressed$;revof$;Fntab$(75,30);INT(average*1000)/1000;" ";
1530 NEXT letter$,line$
1540 IF accuracy=0 THEN accuracy=(right%/(right%+wrong%))*100 ELSE
accuracy=(accuracy+(right%/(right%+wrong%))*100)/2
1550 PRINT curof$;
1560 RETURN
1570 ;
1580 REM ----- File name -----
1590 PRINT Fntab$(10,20);"File name:";SPACE$(60);
1600 found%=0
1610 found$=FIND$("%,TYP",found%+1)
1620 IF found$="" THEN 1660
1630 found%=found%+1
1640 PRINT Fntab$(72,16+found%*2);LEFT$(found$,8)
1650 GOTO 1610
1660 PRINT Fntab$(21,20);curown$;
1670 LINE INPUT "",file$
1680 PRINT curof$
1690 RETURN
1700 ;
1710 REM ----- Load progress -----
1720 GOSUB 1590
1730 IF FIND$(file$+",TYP")="" THEN GOTO 1720
1740 OPEN "I",1,file$+",TYP"
1750 INPUT # 1,average,accuracy
1760 FOR key%=1 TO 83
1770 INPUT # 1,need%(key%)
1780 NEXT key%
1790 CLOSE
1800 RETURN
1810 ;
1820 REM ----- Save progress -----
1830 GOSUB 1590
1840 IF FIND$(file$+",TYP")<>"" THEN GOTO 1830
1850 OPEN "O",1,file$+",TYP"
1860 WRITE # 1,average,accuracy
1870 FOR key%=1 TO 83
1880 WRITE # 1,need%(key%)

```



```

1890 NEXT key%
1900 CLOSE
1910 RETURN
1920 ;
1930 REM ----- Draw keyboard -----
1940 FOR key%=1 TO 82
1950 READ key$,len%,x%,y%
1960 key$(key%)=key$;len%(key%)=len%:x%(key%)=x%;y%(key%)=y%
1970 x%=x%+6
1980 IF letter$(key%)="" THEN letter$=SPACE$(len%(key%)) ELSE
letter$=letter$(key%)+ " "
1990 IF key%=33 THEN GOSUB 2110:GOTO 2010
2000 GOSUB 2050
2010 NEXT key%
2020 RETURN
2030 ;
2040 REM ----- Normal key -----
2050 PRINT Fntab$(x%,y%);CHR$(150);STRING$(len%,154);CHR$(156)
2060 PRINT Fntab$(x%,y%+1);CHR$(149);letter$;CHR$(149)
2070 PRINT Fntab$(x%,y%+2);CHR$(147);STRING$(len%,154);CHR$(153)
2080 RETURN
2090 ;
2100 REM ----- Return key -----
2110 PRINT Fntab$(x%,y%);CHR$(150);STRING$(4,154);CHR$(156)
2120 PRINT Fntab$(x%,y%+1);CHR$(149);STRING$(4,32);CHR$(149)
2130 PRINT Fntab$(x%,y%+2);CHR$(147);CHR$(156);STRING$(3,32);CHR$(149)
2140 PRINT Fntab$(x%+1,y%+3);CHR$(149);STRING$(3,32);CHR$(149)
2150 PRINT Fntab$(x%+1,y%+4);CHR$(149);STRING$(3,32);CHR$(149)
2160 PRINT Fntab$(x%+1,y%+5);CHR$(147);STRING$(3,154);CHR$(153)
2170 RETURN
2180 ;
2190 DATA ,1,2,3,4,5,6,7,8,9,0,-,=,.,,.,.,
2200 DATA ,Q,W,E,R,T,Y,U,I,O,P,I,J,.,.,.,
2210 DATA ,A,S,D,F,G,H,J,K,L,;,<,>,.,.,
2220 DATA ,Z,X,C,V,B,N,M,"",",",.,.,?,@,.,.,.,
2230 DATA ,.,+,,,-,.,.,.,.,
2240 ;
2250 DATA "",2,0,0,"!",2,4,0," ",2,8,0,"3f",2,12,0,"4$",2,16,0
2260 DATA "5%",2,20,0,"6'",2,24,0,"7&",2,28,0,"8*",2,32,0,"9(",2,36,0
2270 DATA "0)",2,40,0,"-_",2,44,0,"=+",2,48,0,"",2,52,0,"",2,56,0

```

```

2280 DATA "",2,60,0,"",2,64,0,"",2,68,0,"",2,72,0
2290 ;
2300 DATA "",4,0,3,"qQ",2,6,3,"wW",2,10,3,"eE",2,14,3,"rR",2,18,3
2310 DATA "tT",2,22,3,"yY",2,26,3,"uU",2,30,3,"iI",2,34,3,"oO",2,38,3
2320 DATA "pP",2,42,3,"[{" ,2,46,3,"}]",2,50,3,"",4,54,3,"",2,60,3
2330 DATA "",2,64,3,"",2,68,3,"",2,72,3
2340 ;
2350 DATA "",5,0,6,"aA",2,7,6,"sS",2,11,6,"dD",2,15,6,"fF",2,19,6
2360 DATA "gG",2,23,6,"hH",2,27,6,"jJ",2,31,6,"kK",2,35,6,"lL",2,39,6
2370 DATA ";;",2,43,6,"$<",2,47,6,"#>",2,51,6,"",2,60,6,"",2,64,6
2380 DATA "",2,68,6,"",2,72,6
2390 ;
2400 DATA "",7,0,9,"zZ",2,9,9,"xX",2,13,9,"cC",2,17,9,"vV",2,21,9
2410 DATA "bB",2,25,9,"nN",2,29,9,"mM",2,33,9,"",2,37,9,"",2,41,9
2420 DATA "/?",2,45,9,"k0",2,49,9,"",5,53,9,"",2,60,9,"",2,64,9
2430 DATA "",2,68,9,"",2,72,9
2440 ;
2450 DATA "",4,0,12,"",4,6,12,"",2,12,12,"",30,16,12,"",2,48,12
2460 DATA "",2,52,12,"",2,56,12,"",2,60,12,"",2,64,12,"",2,68,12
2470 DATA "",2,72,12
2480 ;
2490 DATA 1,"The quick brown fox jumped over the lazy dog."
2500 ;
2510 DATA 1,"She sells sea shells, by the sea shore."
2520 ;
2530 DATA 2,"I saw Esaw sawing wood, and Esaw saw I saw him."
2540 DATA "Though Esaw saw I saw him saw, still Esaw went on sawing."
2550 ;
2560 DATA 5,"To be, or not to be, - that is the question:-"
2570 DATA "Whether 'tis nobler in the mind to suffer"
2580 DATA "The slings and arrows of outrageous fortune,"
2590 DATA "Or to take arms against a sea of troubles,"
2600 DATA "And by opposing end them?"
2610 ;
2620 DATA 5,"His flashing eyes, his floating hair!"
2630 DATA "Weave a circle round him thrice,"
2640 DATA "And close your eyes in holy dread,"
2650 DATA "For he on honey-dew hath fed,"
2660 DATA "And drunk the milk of Paradise,"
2670 ;

```

```
2680 DATA 2,"But, soft! what light through yonder window breaks?"
2690 DATA "It is the east, and Juliet is the sun!"
2700 :
2710 DATA 5,"ABS ADDKEY AND AS ASC ATN BUFFERS CALL CHAIN CHR$ CLOSE"
2720 DATA "COS DATA DELETE DIM ELSE ERA EXP FIND$ FOR GET GOSUB GOTO"
2730 DATA "HIMEM IF INKEY$ INPUT# INT LEN LOAD LPRINT MAX MOD NEXT"
2740 DATA "NOT ON OPEN PEEK POKE RENUM RETURN RUN SAVE SIN SPACE$"
2750 DATA "STEP TAB TAN THEN TO VAL WAIT WEND WHILE WRITE ZONE"
2760 :
2770 DATA 2,"How much wood could a wood-chuck chuck, if a wood-chuck
could chuck wood?"
2780 DATA "He'd chuck, he would, as much as he could, if a wood-chuck
could chuck wood."
```


9

Commercially Available Software

WHAT SOFTWARE?

When new computers are introduced, they usually lack software. This inhibits sales because most people want their computers to *do* things, and so prefer to buy those with plenty of programs. This in turn discourages software houses from writing programs because there isn't a large enough user base to make it worthwhile. So it becomes a vicious circle that usually takes about a year to sort itself out. No such problems occurred with the Amstrad word processor – because it uses the longstanding operating system CP/M, many programs were already in existence when the machine was introduced. All that was needed was to *install* them for the Amstrad and distribute them on 3-inch discs. Apart from other word processing programs, there are programs to help you keep accounts, spreadsheets for financial and statistical modelling, graphics packages to create diagrams and graphs, and other computer languages.

WHAT WILL IT COST?

There seems to be a rule of thumb in the computing industry regarding the price of software. This is based on *what the market will stand*. This seems to be about 10% of the cost of the hardware. Up until recently machines like the Amstrad word processor would have cost one or two thousand pounds. As a result, many software packages cost £200 or more. With the PCW8256 costing about £450, most software packages seem to cost around £49.99, even if exactly the same product costs many times more for a more expensive computer. So, not only is the Amstrad excellent value as a machine, but the software is very good value too. Sooner or later the unfortunate

owners of those other machines will begin to realise what's going on, and press for lower prices all round.

ALTERNATIVE WORD PROCESSORS

Having bought an Amstrad complete with LocoScript, who would want to spend more money on another word processor program? Well, there could be a reason, although it would have to be a good one. Some people may already be familiar with another CP/M word processor such as WordStar, and not want to learn a new one. If LocoScript has a fault, it is that it scrolls very slowly through long documents. One answer to this is to keep the length of documents down by splitting the text into smaller parts. If you *do* handle long documents, another program might be preferable. If you want to use a different printer to the one supplied, a daisywheel perhaps, then you will need a word processor program that works under CP/M (until some enterprising person markets a box that allows you to plug the other printer into the normal printer outlet). Some people who do a lot of data communications prefer to have a word processor that creates ASCII files as normal and works under CP/M, so that they can go straight to the communications program rather than reload the computer from scratch. And finally you may require features that LocoScript doesn't have (yet – at the time of writing – but may do eventually). These could be mail-merge, where a mail-list could be combined with word processing to automatically produce 'individualised' letters, and a spelling checker. This is a program which compares the words in a word processed document with a dictionary of words stored on a list. Mis-spelt words won't be on the dictionary disc, and are brought to your attention for correction if necessary. Of course, just because a word doesn't appear in the dictionary doesn't automatically mean that it's mis-spelt, it may be a specialist word, or colloquial or a name. Unusual words that you use regularly can usually be added to the dictionary, or at least, a supplementary list. Spelling checkers can't check for words that exist in the dictionary, but are used in the wrong context in the document – *there* and *their* – for example. But on the whole they can be a great help, provided that you don't depend on them entirely.

So you can see that there are several reasons why you *might* want an alternative word processing program.

The best known CP/M word processing program is **WordStar**, available on CP/M machines for many years. Another program, which is rather similar in operation to WordStar is **New Word**. This is available for the Amstrad. In WordStar and New Word, the formatting and editing commands are entered using *control characters*, that is by holding down the <ALT> key together with a letter key. Until the user learns these, a help menu can be displayed across the top half of the screen. This does reduce considerably the amount of screen available to display the document being edited. In fact there are three levels of help menu to match different user familiarity. Even at the lowest help level, full help menus can be called up when required.

These programs work under CP/M, have the ability to scroll quickly through long documents and produce ASCII files as standard. For these reasons it's quite popular with many people who use electronic mail, want to edit large documents quickly, or want to send files to other computers in ASCII. New Word is supplied complete with a mail merge facility and a spelling checker program. This latter has a dictionary of 11,000 words, and you can create a supplementary file of words that you use regularly that aren't contained in the main list. It also includes an excellent routine which will search through the dictionary looking for any words that match a string of letters and wildcards entered at the keyboard. For example, if you enter:

?O?P???R

it will find:

**COMPARER, COMPILER, COMPOSER, COMPUTER, and
GOSPELER** (it's an American dictionary).

This little program alone is worth its weight in gold to people like me who like to cheat at crosswords!

As this book goes to press, two spelling checkers have become available which work with LocoScript files.

DATABASE PROGRAMS

Any organisation (and many individuals) depend on information. That information might be about orders received from customers, or invoices for goods delivered. It might be about the hours worked by its employees. This is administrative information. Manufacturing

companies need other sorts of information too, for instance how it makes its products, where it gets its components from, and the properties of the materials that it uses. On a more personal level, you might want to keep an address list of all your friends and contacts, or a list of your gramophone record collection. All this information needs to be stored for future reference. What is most important is that it is stored in such a way that each piece of information can be easily found again when required. The traditional way of storing information is to file it. That is, to classify it according to the type of information, and then store it in a suitable way. Filing cabinets are usually used for documents and correspondence, card files for record cards, and plan chests for large drawings.

Supposing that you run a small business and you need to keep a record of all the items that you keep in stock. You might use record cards for this purpose. Each card would have a standard layout so that you know where to store and look for each item of information. These might be for the type of item, the supplier (so that you know where to order more), the usual cost, the selling price, the minimum stock that you keep before re-ordering and the usual time taken to supply. The space reserved for each item is known as a **field**. The whole card forms a **record**.

All the cards are now stored in a **file**. So these are the three elements of any storage and retrieval system:

field
record
and **file**.

There may be many hundreds of different records in one file, and so that any one card can be found easily they are stored in a particular order. But what order is best? Do you store them in order of the name of item? Or would it be better in types: all the bolts together (if it's a hardware business), all the nuts and so on? Maybe you will want to review all the items that you get from one particular supplier.

The problem with pieces of paper and pieces of card is that they can only be stored in one particular order, so they can be difficult to retrieve for other classifications. If you want to cross-reference them, it will involve extra copies of the cards, and that can make difficulties if the cards need to be updated.

Computers can now be used to replace the more traditional methods of information storage and retrieval. For this purpose, special computer programs called *database* packages have been developed. These can have significant advantages over the old systems.

The items of information are kept on floppy disc, and can be stored and retrieved from the keyboard. Using a computerised database, it no longer matters in what order the records are stored. The computer can search through the entire contents of the disc to match given parameters for items or records, or even in some cases to look for a given word. So, for instance, you may instruct the computer to present you with all the records which show components from a given supplier that go into a particular product and that usually take more than three weeks to deliver.

The **PCW-INDEX** program listed in Chapter 8 is a simple example of a database. Others, available commercially, will have more flexible formatting, and more powerful search facilities. For example, if a field contains a number, such as price, you will be able to search for all items above a certain value. Some packages allow you to combine a mail-list with text for a mail shot. Using **PCW-INDEX**, however, will give you some valuable experience and enable you to judge which features are important to your application *before* you spend money on another package.

SPREADSHEETS

Much of the time spent by office workers is used to make decisions, and it's very helpful if a way can be found to calculate the possible effects of some action that might be taken. As with all activities, there are many inter-related factors which each have an effect on the final result, so a way is needed to take all these into account.

There is a type of computer program which helps to monitor and control inter-relating numbers, and these can be used for many different purposes: accounting, scientific calculation, industrial and engineering applications. These programs are called *spreadsheets*. They handle sets of figures and make very rapid calculations of a sort which would take a long time to do on paper.

To see how a spreadsheet works, we'll look at an example of the sort of task for which it could be used. Let's imagine doing a set of

calculations on paper – let's calculate the cost of running a car for a year.

To find out what it costs to run a car we need to take a lot of different factors into account. The most obvious is the amount of **petrol used**. This of course will be very much dependent on the **number of miles** driven, and the number of miles that the car will do to the gallon – the petrol **consumption**. We'll assume that petrol costs two pounds a gallon.

Then there are two fixed costs – the **road fund tax** and the **insurance**. These are the same irrespective of the mileage.

Then there is the **depreciation** of the car, that's the amount that it loses in value over the year. Finally we will take into account the **lost interest**. The car cost a certain amount of money to buy. If, instead of buying the car, that money had been invested, perhaps in a building society, then it would have earned interest. Let's assume that the interest rate is eight percent, then the money would have earned eight percent of the purchase price in a year. Because we bought a car instead of investing the money, we don't get the interest, so it must be taken into account as a cost.

The total cost isn't a simple total of all the factors, some calculation has to be done before the actual costs can be worked out.

- [0] **RUNNING COSTS.**
- [1] **mileage**
- [2] **consumption**
- [3] **petrol cost ([1]÷[2]x£2)**
- [4] **road fund tax**
- [5] **insurance**
- [6] **purchase price**
- [7] **resale value**
- [8] **depreciation ([6] - [7])**
- [9] **lost interest ([6]x.08)**
- [10] **total cost ([3]+[4]+[5]+[8]+[9])**

So here we have a financial model on which to calculate the costs of running a car. Of course it's nowhere near complete because there are a lot of things that we still haven't included, like servicing. But it will give us an idea of the way that we can use a spreadsheet program to calculate the effects of several factors.

Some of the items on our list, like the road fund tax, can be added directly to the total running costs. Some, like the mileage, are factors which are used to calculate an item of the cost.

Now we'll add some actual figures to see how the calculation will work out. Let's suppose that the car is a Mini, which cost £4000 to buy, and will be worth just £3500 if we want to sell it at the end of the year. The petrol consumption is 40 miles to the gallon and it's driven for ten thousand miles. Road fund tax is, say, £100, and insurance another £100. We now have our basic factors.

[0]	RUNNING COSTS.	
[1]	mileage	10000
[2]	consumption	40
[3]	petrol cost $([1] \div [2] \times £2)$	
[4]	road fund tax	100
[5]	insurance	100
[6]	purchase price	4000
[7]	resale value	3500
[8]	depreciation $([6] - [7])$	
[9]	lost interest $([6] \times .08)$	
[10]	total cost $([3] + [4] + [5] + [8] + [9])$	

We can now work out items 3, 8 and 9 and hence calculate item 10, the total running costs.

[0]	RUNNING COSTS.	
[1]	mileage	10000
[2]	consumption	40
[3]	petrol cost $([1] \div [2] \times £2)$	500
[4]	road fund tax	100
[5]	insurance	100
[6]	purchase price	4000
[7]	resale value	3500
[8]	depreciation $([6] - [7])$	500
[9]	lost interest $([6] \times .08)$	320
[10]	total $([3] + [4] + [5] + [8] + [9])$	1520

Supposing though, you wanted to see how much it would cost if you only ran the car for five thousand miles. The answer wouldn't be half as much, because a lot of the costs, such as the road fund tax, are fixed. To get the answer, you would need to change item one, item three, and of course item ten.

In a spreadsheet program, once you have changed item one and pressed the <ENTER> key, the program would calculate all the other figures automatically and instantly. And it would be able to handle much more complex tasks than the simple example that we've seen.

A spreadsheet is a computer application program, which comes ready for the user to insert his own data. It's rather like a grid. In the example below, you see there are only two columns and ten rows. In a real package it will be possible to use far more, perhaps two hundred and fifty six columns by two hundred and fifty six rows. The individual positions in the columns and rows are known as cells. Each cell is known by its grid position, and in our example cell [b6] contains the figure 4000.

[a]	[b]
[1] mileage	10000
[2] consumption	40
[3] petrol cost	500
[4] road fund tax	100
[5] insurance	100
[6] purchase price	4000
[7] resale value	3500
[8] depreciation	500
[9] lost interest	320
[10] total	1520

Each cell on the spreadsheet can contain information of one of three types – a **label**, a **constant** or a **formula**. Our example shows all three. All the cells in column [a] are **labels**. These can be added to make the spreadsheet more readable. Cells [b1], [b2], [b4], [b5], [b6] and [b7] contain **constants**. A constant is a figure entered by the user, and not affected by anything else. Cells [b3], [b8], [b9] and [b10] however contain **formulae**.

The formula for [b3] is the contents of cell [b1] divided by the contents of cell [b2] and that result multiplied by 2 to give the cost of the petrol used in pounds. If the contents of one of the cells used as an element in the formula is changed, then the spreadsheet program automatically calculates the effect throughout. The formula for [b8] is [b6] minus [b7]. For [b9] it's [b6] times nought point nought eight (to give us eight percent). And finally for [b10] the total cost is the

sum of [b3], [b4], [b5], [b8] and [b9]. So, if we change the number of miles that are driven in the year to five thousand, then the program updates all the cells affected.

The spreadsheet can be used not only to calculate what *has* happened, but also to see what *might* happen if certain factors change. This sort of application is known as a **what if** application. Spreadsheets are so easy to use that it is easy to calculate what will happen *if* such and such occurs. The spreadsheet can also be used to consider comparisons. Here our example is being extended to make a comparison between running a Mini and a Rolls Royce.

[a]	[b]	[c]
[0] RUNNING COSTS.	MINI	ROLLS
[1] mileage	10000	10000
[2] consumption	40	10
[3] petrol cost	500	2000
[4] road fund tax	100	100
[5] insurance	100	500
[6] purchase price	4000	25000
[7] resale value	3500	25000
[8] depreciation	500	0
[9] lost interest	320	2000
[10] total	1520	4600

To make the program easier to use, some cells can be made to be the same as others. Here, cell [c1] contains the formula [b1]. This simply copies the contents of [b1] so that if you change the mileage in [b1], then the value held in [c1] changes too, and so the total cost for both types of car is calculated automatically, so that a comparison can be made. Here are the contents of the cells in detail.

[a]	[b]	[c]
[0] label	label	label
[1] label	constant	[b1]
[2] label	constant	constant
[3] label	[b1/b2x2]	[c1/c2x2]
[4] label	constant	[b4]
[5] label	constant	constant
[6] label	constant	constant
[7] label	constant	constant
[8] label	[b6-b7]	[c6-c7]

[9] label	[b6x0.08]	[c6x0.08]
[10] label	[b3+b4+b5+b8+b9]	[c3+c4+c5+c8+c9]

Several different spreadsheets are available for the Amstrad. The software is usually supplied together with extensive instruction manuals. Although the software is ready to use, as with any tool, software or hardware it must be set up to do any particular task.

Supercalc 2 is a well established spreadsheet package that has been around for some time on other CP/M machines and is now available for the Amstrad. A newer program, the Cracker, allows the output to be in the normal tabulated form, and also in the form of graphs and charts for easier interpretation. In this way, the same software package can be used for many different purposes.

ACCOUNTS PACKAGES

There are now many accounting and payroll packages available for the Amstrad as this is the ideal low cost machine for the small business. This is a specialist field and the best advice is to go to your dealer and ask to see the packages demonstrated, in order to find which one suits your needs the best.

GRAPHICS PACKAGES

Two graphics packages for the Amstrad word processor are DR Draw and DR Graph, both from Digital Research, the company that created CP/M. These programs allow you to prepare diagrams and graphs on the screen of the Amstrad, and then print them out on the printer. Both these packages run under CP/M (as might be expected). Both consist of several utility programs and make use of the memory drive to hold them. There are precise instructions on how to create working discs from those supplied. The instructions assume that you only have one floppy disc drive fitted, if you have two, then copy the two sides of the disc supplied onto *two* working discs. The *side B* working disc must be put in the lower disc drive. The utilities on this are copied into the memory drive, and so it can eventually be removed.

DR Draw

DR Draw puts an easel on the screen for the user to create diagrams

on. This is done by moving a cross-hair around the drawing surface using the cursor control keys. There are facilities for drawing lines, bars, circles, etc. As each feature is added, it is recorded to the disc. Features can subsequently be removed if required. Areas can be filled with various forms of shading. Graphics so created can also be stored on disc, and then recalled and edited later. Given patience and the will to succeed it's possible to create good clear diagrams with this package. They can be printed out onto paper and used as illustrations for reports, or for overhead projection transparencies. DR Draw is more likely to please a technical person who wants to produce clear diagrams, than an artistic person who wants to create images through a computer.

DR Graph

DR Graph is used in a different way to DR Draw. In this case graphs are produced from data entered through screen menus. A wide range of different graphs can be produced, including **bar charts**, **line graphs** and **pie charts**. It's even possible to incorporate several different types of graph on the same page. Figure 9.1 is a sample illustration from DR Graph.

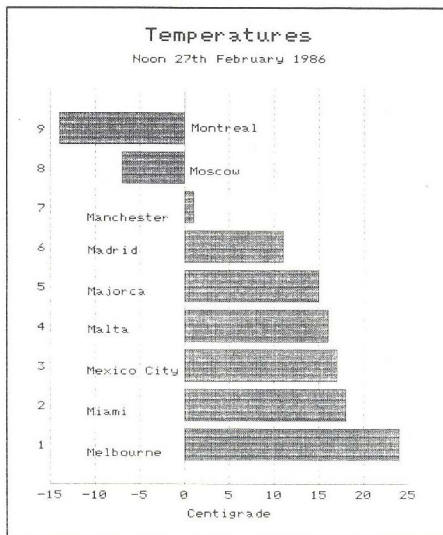


Figure 9.1 Sample Illustration from DR Graph

COMMUNICATIONS PROGRAMS

If you do a lot of data communications, you may want to look for a more sophisticated terminal emulation program than **mail232** which is supplied with the Amstrad. If you want to access **viewdata** services, then you will certainly need another program, because **mail232** won't work for these. One communications program which has previously been available for other micros, and has now been released for the Amstrad, is **Chit-Chat** by Sagesoft. This is available for **electronic mail** (like **mail232**), and **viewdata**, or as a combined package with both programs together. Of course you will need a serial interface (see Chapter 10), and a suitable modem. If you don't already have these, you can buy them packaged with the software (other similar packages are available from other software suppliers).

Chit-Chat is a very comprehensive and professional package. It has a directory screen which can contain details of the services that you will want to contact. The number that can be held is limited only by the space available on the floppy disc. As well as the telephone numbers of the services, the directory also holds details of the necessary terminal parameters and log-on procedures for each. These must be entered into the directory through a menu type screen correctly before you access the service. Once entered however, this information is recorded for future use. There are already entries in the directory for about 30 popular services, although some of these may need some alteration depending on the area that you are calling from. There is a comprehensive manual which explains each point very thoroughly.

Once the entries are in the directory, **Chit-Chat** is very easy and convenient to use. If you have an auto-dial modem, **Chit-Chat** will be able to drive this, and to perform tasks unattended. This could be to transmit long files late at night when the telephone system is available at cheap rates for example.

As well as electronic mail, a version of **Chit-Chat** is available which can handle **Viewdata**. **Viewdata** services usually rely rather heavily on colour and screen graphics. The Amstrad word processor can't provide these; the monitor is monochrome, and the special **Viewdata** characters really need a dedicated terminal. So how well does **Chit-Chat** perform against these difficulties?

There is a special character set which copes very well with Viewdata graphics. There are only two colours of reproduction however, black and green, so the Viewdata colours have to be reproduced in one or other of these. The result is that in some cases where text is meant to be shown against a coloured box, both text and box come out in the same shade on the Amstrad's screen, and are unreadable. In most cases though, important text on information pages is shown against black and so is reproduced well. Figure 9.2 shows a sample Viewdata screen using Chit-Chat.



Figure 9.2 Viewdata Screen using Chit-Chat

At the time of writing, several other software houses had communications packages for the Amstrad in the pipeline. Packages such as these are a relatively easy way to get into data communications; they should include all that you need (although you may need to buy the *serial interface* separately), and the instructions will cover the modem and the software.

WHERE TO LOOK FOR SOFTWARE

Your local computer shop will probably carry a lot of software for the Amstrad Word Processor and should be willing to let you see it in operation in the shop, so that you can judge if it's what you need. There are many magazines that carry articles and reviews of what's currently available. Two general (and fairly serious) computing magazines that are usually worth looking at are **Personal Computer World** and **Practical Computing**. Of more particular interest to Amstrad owners are two magazines which specialise in the Amstrad Word Processor (and the Amstrad Personal Computer 1512). They are **Amstrad Professional Computing** and **8000 Plus**. As these two publications cater for two models of Amstrad computers, you must make sure that the software that you are looking at will run on the PCW. With a lot of Amstrad word processors about, a lot of new software will be coming onto the market, so look around carefully before making up your mind.

10 What Are Data Communications?

INTRODUCTION

You may have heard of some of the recent advances in data communications, such as electronic mail. Your Amstrad can be used as a data communication terminal, but to see if it's of any potential interest and value to you, you need to be able to answer these questions. What are data communications, who uses them and who needs to understand them?

Data communications (*data comms* for short) concerns the transmission of data, that is computerised information, between computers.

Who uses it? Well the short answer to that is that although they may not realise it, practically everyone in the western world does, except hermits! Nearly every aspect of modern life involves data transmission of some sort or other. That's quite amazing considering that 30 years ago data communications as such didn't exist, but then neither did computers as we know them now. The nearest thing to *data comms* were the telex and the telegram, used only for very urgent or special occasions (most people regard the latter only as the bringer of bad news!). But now if you have a bank account, if you rent a television set, if you buy goods in a large store, or if you do one of a hundred and one other things in everyday life, then you are using services which rely on data communication. Forty percent of all working people handle not goods, but *information*.

And that leads us on to the answer to the third question posed above – who needs to understand them? Because we are all affected

by data communications, we all really ought to know something of what they're all about.

What better way of finding out about data communications than using them for yourself? With the ordinary telephone in your home you can call up almost any one of the estimated 500 million other people connected to public telephone networks around the world. If you have an Amstrad, the additional CPS8256 Centronics/serial interface adaptor and a modem, you have the means of communicating with other computers anywhere where there is another telephone. There should be enough scope there for anyone.

Communicating with another computer doesn't just mean sending messages to some other computer enthusiast just for the sake of doing it. It can mean sending and receiving electronic mail, or accessing databases to get useful up-to-date information when you need it.

In this section we will be exploring some of the ways that you can use your Amstrad to communicate with other computers. Data communications is a very large subject, and people need to spend a long time studying it and working with it to become experts. You don't need to be an expert to get a lot of interest and enjoyment out of using your Amstrad to communicate with others. But you will need to understand some of the basic principles of what is involved, and that is what this part of the book is designed to help you to do. You will find out what a modem is and what types are available. The different standards for transmission, and transmission speeds will be explained.

For data communications to take place between computers, there are two separate sets of requirements to be met. These are *hardware* and *software*.

HARDWARE

To use data communications you need a serial port. This allows you to send data to, and receive data from, outside devices. There is an optional serial port available for the Amstrad known as the **CPS8256 Centronics Parallel/RS232 Serial Interface** (see Figure 10.1). As its name suggests, it also allows data to be sent to a Centronics type printer. Communication further afield means the use of the telephone. This needs a modem, and these are described later.

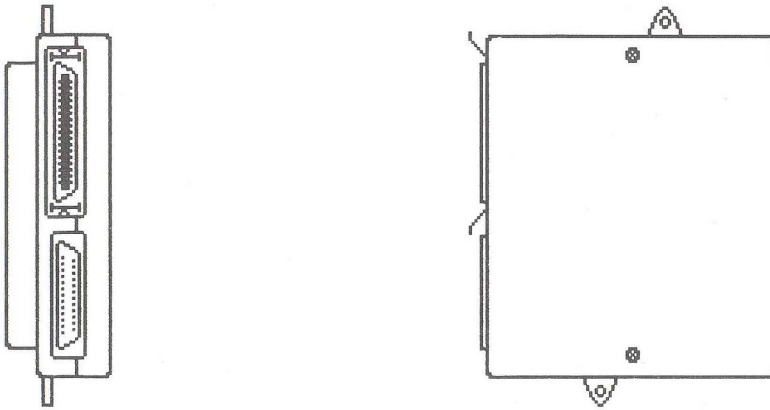


Figure 10.1 CPS8256

SOFTWARE

With most other personal computers you have to buy communications software costing at least £30. A communications program called **MAIL232** actually comes *free* when you first buy the Amstrad, although it's hidden away on the LocoScript disc and not documented in the basic manual. This will allow communication between computers and access to most bulletin boards, electronic mail and database services, but not viewdata services such as Prestel and Micronet. For details of a viewdata software package see Chapter 9.

In this chapter I will explain the terms that tend to baffle newcomers to data communications. This will help you to decide what equipment is necessary to access the bulletin boards or on-line services that interest you most.

DATA

Computers deal in **numbers**, everything is handled that way. Even in the word processing program, all the characters are handled by the Amstrad as numbers. The numbers in turn are broken down to ones and zeros and these are handled and stored as the presence or absence of voltages. The basic building block of computer data is the **bit**. The Amstrad word processor is an *eight-bit* computer, that is it handles data eight bits at a time (eight bits are known as a **byte**). This

is possible because the data buses, that is the tracks on the printed circuit board carrying the data from the memory to the processor, etc, are arranged as eight parallel circuits. Thus, within the Amstrad's main unit, each byte is sent, all eight bits at once.

An eight-bit computer can handle numbers ranging from 0 to 255. Each bit within a byte has a certain value, depending on its position in the byte. Each bit can be in one of two states, either a high voltage (usually 5 volts) or a low voltage (0 volts). This is usually represented as a 1 or a 0 (1 for high voltage and 0 for low voltage). As the bits emerge from the serial output port of the micro, the first bit (after the start bit) is the least significant bit representing the decimal value of 1. The next represents the value of 2, the next of 4 and so on until the eighth represents the value of 128. If all eight bits are high, then the combined value will be:

$$\begin{array}{cccccccc} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1+ & 2+ & 4+ & 8+ & 16+ & 32+ & 64+ & 128 = 255 \end{array}$$

So 256 different values (including 0) can be represented within the computer. Here is an example of how a different value would be held in a byte:

$$\begin{array}{cccccccc} 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1+ & & 4+ & 8+ & 16+ & 32 & & = 61 \end{array}$$

SERIAL TRANSMISSION

Sometimes we need to send data out from the computer. This may be over short distances, to a printer for instance, or else over a long distance, to a remote computer perhaps. Over short distances there is no problem in extending the parallel tracks along a ribbon cable. The ribbon cable to the printer is an example of this. However, if we wish to use this method for data transmission over long distances, then we would need at least eight telephone lines all working at once!

Understandably, a different arrangement – known as serial transmission – is used. Serial transmission is also used for short distances: some printers use this method, and so for that matter does the Amstrad word processor keyboard, but that is rather more special. Here, the eight bits are sent one after the other down a single line. To achieve serial transmission, each byte is loaded into a buffer, then each bit is sent out in turn. The least significant bit (representing a value of 1) goes first, followed by that representing 2 and so on.

One bit looks very much like another, and the receiving computer needs to be able to tell when one byte stops and another one starts. For this, two extra bits are added to each group of eight – a start bit and a stop bit. So in data communications, the word length is usually ten bits.

The information leaves the computer through the serial port, marked on some computers as the RS232 port, and on others (notably the BBC micro) as RS423 (which is substantially the same). Serial information can also be fed *into* the computer through the same connection.

STANDARDS

If different pieces of equipment are to be joined together and be able to understand each other, then it is necessary for them to work in the same way. To achieve this, various bodies have proposed standard ways of working. Two standards are of interest to us in this chapter, ASCII and RS232.

ASCII (pronounced ‘askey’)

This acronym stands for the *American Standard Code for Information Interchange*. It covers the way that the bit patterns in each byte are interpreted and is used in almost all computers. In the ASCII code, some of these values are taken to represent different characters, either letters of the alphabet or numbers, or else control characters. However for the purpose of normal data communication, only seven bits are used, allowing just 128 different characters or codes. These are quite enough to represent all the letters of the alphabet in both upper and lower case, as well as the numbers 0 to 9 and plenty of codes. These are shown in Appendix 2.

During data transmission, the word length is usually ten bits. Allowing for the start bit and the stop bit for each word, this leaves an extra bit (the eighth). In some cases, the eighth bit is used for error checking at the receiving end, and this is discussed below under Parity, but Viewdata systems use it to send additional information.

Viewdata transmission requires more codes, as colour and graphics information is sent, and these codes are sent using the eighth

bit. MAIL232 doesn't support Viewdata, although suitable software is available (see Chapter 9). Most Viewdata services such as Prestel and Micronet rely very much on colour.

RS232 (RS423)

RS232 is a standard connection used between computers and peripherals, or between computers and computers. The RS232 attachment for the Amstrad uses a 25-way D connector. A 'female' plug is needed on any cable to be connected to it.

ERROR CHECKING

Data communication over long distances (and over short ones if the telephone connection isn't of good quality) is prone to disturbance causing errors to get into the message. Many systems use methods whereby the incoming data is checked. If an error is detected, a request can be returned to the sending computer for the corrupted data to be repeated.

Only values up to 127 are used for normal ASCII characters, which can be achieved using only 7 bits, leaving us with a spare bit in the normal word length of ten. This can be used to detect errors using a method known as parity checking. There are two types of parity, even parity and odd parity.

Even Parity

The sending computer checks each byte and counts the number of 1s. It then adds an eighth bit, ensuring that there are an even number of 1s. A start and a stop bit are then added so that there are 10 bits altogether and this is transmitted.

The receiving micro strips the start and stop bits off again and then counts the number of 1s in the remaining byte. If that number is even, the byte is assumed to be correct. Of course this system isn't foolproof, but it's much better than nothing.

Here are seven bits representing the value of 93:

bit7	bit6	bit5	bit4	bit3	bit2	bit1
1	0	1	1	1	0	1
64+		16+	8+	4+		1

Here there are five 1s, an odd number, so the bit added to check parity is a 1, ie:

bit8							
	1	1	0	1	1	1	0
							1

Here are seven bits representing the value of 34:

bit7	bit6	bit5	bit4	bit3	bit2	bit1
0	1	0	0	0	1	0
	32		+		2	

Here there are two 1s, an even number, so the parity bit is a 0, ie:

bit8							
	0	0	1	0	0	0	1
							0

Odd Parity

With odd parity, as you can imagine, bit eight is used to ensure an odd number of 1s. Many communication systems ignore parity altogether, those which do use it usually look for even parity.

TRANSMISSION

Baud Rate

The speeds at which the computer will send out and receive data are critical. Most computers will run at a range of speeds, but the transmission equipment may limit those which can be used.

The transmission speed is usually referred to as the **Baud rate**. Named after the French communications pioneer, Emile Baudot, this roughly refers to the number of *bits* transmitted per second. As each character usually needs ten bits to describe it, then the Baud rate divided by ten will give the number of characters transmitted per second. The Amstrad word processor together with the CPS8256 serial adaptor will support baud rates of 50, 75, 110, 134.5, 150, 300, 600, 1200, 1800, 2400, 3600, 4800, 7200, 9600 and 19200. Transmission and reception can either be at the same rate, or at different rates. The speeds *must* be set to agree with those used on the equipment being communicated with.

Duplex

Another term that is frequently used in connection with data trans-

mission is **duplex** (or full-duplex) and **half-duplex**. In full-duplex systems, which are probably the only ones that you are likely to come across, transmission and reception can take place simultaneously over the communications link. In a **half-duplex** system, the computer that is transmitting must stop doing so (as part of the protocol) before it can receive anything from the other.

X on/X off

These are characters sent from one computer to another over a data transmission link, to indicate that it is ready or not ready to receive data. Transmission is stopped if the recipient device is unable to handle it (during a disc operation for example). This is also known as handshaking.

Transmission Through a Modem

The telephone network (the **public switched telephone network** or **PSTN**, to give it its full title) has been around for some time, long before electronic computers were thought of. Not surprisingly, it wasn't designed to carry the sort of signals that computers send out. These are *digital* pulses, a *square wave* rapidly changing between two voltages. The telephone network was designed to carry the human voice, which is an **analogue signal** containing many different frequencies. So when we want to send computer signals over the PSTN, we need some device which will change the nature of the signal so that it can be carried for long distances with the minimum of distortion. The required device is a **modulator/demodulator**, usually known as a **modem**. At the sending end the modem has to convert (modulate) the data pulses into **tones** that the telephone network can transmit successfully. At the receiving end it demodulates them back to pulses that the computer can understand. Under the European CCITT system there are two tone frequencies for each direction of travel, one tone to represent the high voltages, and one for the low. As the tones all end up on the same circuit, the modem has to be able to tell whether they are coming or going, so a different set of frequencies is used to cover transmission and reception for both directions of the communication. One computer will be transmitting a set of tones which the other computer will be receiving, so there is the possibility of some confusion as to who uses which set of

tones for what. There is a convention that one set of tones is known as the **originating tones**, and the other as the **answer tones**. Under this convention, the party *originating* the call, that is the one that picks up the phone and dials the number, uses the originating tones.

Standard Transmission Rates

For normal data communications using a modem and the telephone network there is a limit to the speed at which data can be transmitted reliably because of the limited **bandwidth** available. Bandwidth refers to the highest frequencies that the network can carry. For high speed data transmission, special dedicated circuits are available, but these are outside the scope of this book. Using **PSTN**, there are two sets of speeds which are the most common. They are usually referred to as **300/300** (or **V.21**) and **1200/75** (or **V.23**). Most modern modems work at one or both of these speeds. If you have a modem working at both of these speeds you will be able to communicate with almost all the services that you could need. Each has a different application.

300/300 (V.21)

This allows transmission and reception at about 30 characters a second. This rate is commonly used for such purposes as bulletin boards and electronic mail, where there is likely to be an equal amount of data sent in either direction.

1200/75 (V.23)

Using this standard, reception is about 120 characters a second, but transmission is just 7.5 characters in the same time. This standard was originally introduced for viewdata services such as **PRESTEL**. Here very little data is sent to the host computer, often just the output from a television remote control handpiece, but much more data is received. As the low transmission rate can be handled by very low frequencies, high frequencies can be accommodated within the bandwidth available, allowing a high reception speed. As well as being used for viewdata services, **V.23** is also useful in situations where more data is likely to be received back than sent, thus saving telephone time and, in many cases, log-on charges from the host computer. So **1200/75** can be very useful for accessing databases or other services where a lot of text is likely to be downloaded.

BUYING A MODEM

There are now many modems on the market to choose from. There are two distinct types: **direct connect** modems and **acoustic couplers**.

Direct Connect Modems

These plug directly into the telephone network. In the UK this is done through the 'new style' telephone sockets. If you don't have one of these, British Telecom will fit one. It doesn't cost much more to have extra ones fitted at the same time, and it's much more convenient for data communications if there's one next to your Amstrad. Direct connection has the advantage that there is the minimum of interference from ambient noise in the surroundings. On the other hand there has to be a suitable telephone socket available. However, all new telephone installations in the United Kingdom are now fitted with suitable plugs, and it is easy to get your telephone connection upgraded. Telephone services are understandably rather touchy about any electrical equipment that you connect to their lines. It's not just their exchanges that they're worried about, their engineers might get electrocuted if your equipment is faulty. Any modems used in the United Kingdom must carry an approval label with a green spot. Apparatus not approved, and carrying a label with a red triangle shouldn't be connected. Just because a modem is on sale (even advertised widely), don't assume that it's been approved.

Before buying a modem, you should check that it has all the facilities that you are likely to need. Even quite cheap modems now give a choice of baud rates, and you would be advised to choose one that has 300/300 and 1200/75. Some modems give the choice of CCITT and Bell tones. This is much less useful. If you need Bell tones to call a database in the United States, it would be much cheaper in the long run to use Packet SwitchStream at local telephone call rates. In any case you shouldn't use Bell tones over British Telecom, because the tones could upset some telephone exchanges.

Modem Controls

There will be some controls on the front of a direct connect modem (unless it's a very fancy one that's controlled entirely by computer software). There will be a switch for **local test/on-line**. Set it to local

test at all times except during the actual transmission. There will be another switch for **originating** or **answer tones**. If the modem supports different transmission speeds there will be a switch, button or knob for this.

Some modems support **auto-dial** and **auto-answer**. These can be useful but need special software to make them work, so make sure that that is available for the Amstrad. Some modems approved for normal communications aren't approved for use in auto-dial or auto-answer modes which are subject to separate regulations.

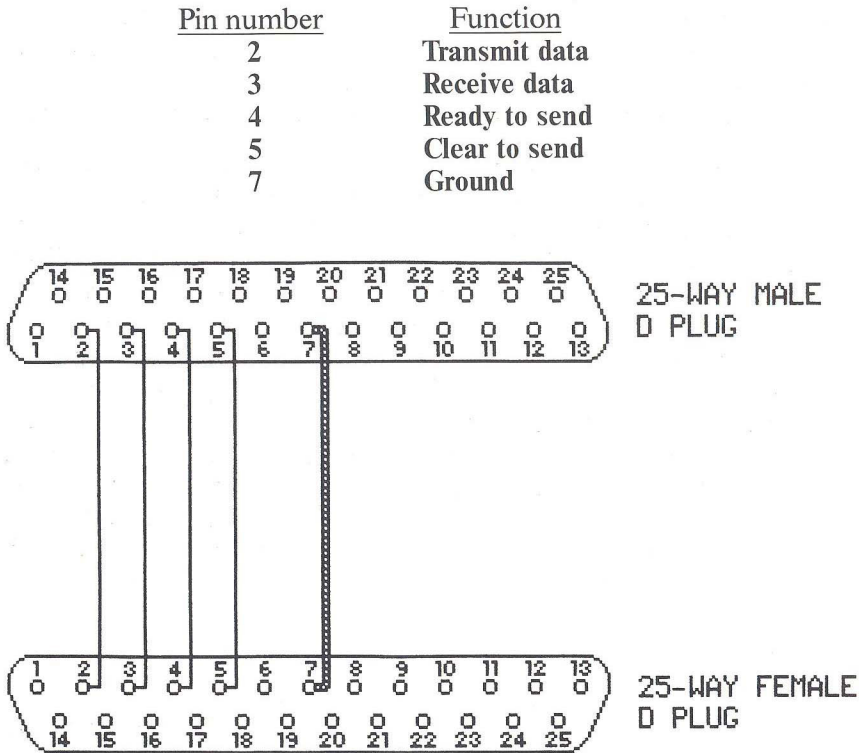
Acoustic Couplers

An acoustic coupler has receptacles for the mouth and ear-piece of the telephone handset. In this way there is no direct electrical connection with the telephone network. This allows the acoustic couplers to be used almost anywhere that there is a telephone. Acoustic couplers only usually work to one set of baud rates. The shape of some modern telephone handsets does not allow a very good connection, permitting extraneous noise to corrupt the signal. In particular 'Trimphones' are almost impossible to use with acoustic couplers.

Acoustic couplers are most useful where you travel around a lot and need to use data communications wherever you happen to be. They can even be used from public telephone boxes. However it's unlikely that you will want (or be able) to use your Amstrad in a telephone booth, so this may not be much of an advantage!

CABLES

The serial output connector on the CPS8256 adaptor is the standard type for RS232. This is a **25-way 'male' D plug**. Most modems will have a 25-way 'female' D socket. So in most cases you will need a lead with a 25-way 'male' D plug on one end and a 25-way 'female' D socket on the other. Only nine of the pins need to be connected. They are pins 2, 3, 4, 5, 6, 7, 8, 20 and 22. In fact for most modems, where you don't need auto-dial and auto-answer, only 5 of the pins need be connected. If you are making up the lead yourself, four-core cable and screen will do, connecting pins 2, 3, 4 and 5 with the wires, and 7 with the screen, as shown below. Figure 10.2 shows these modem cable connections.



MODEM CABLE

Figure 10.2 Modem Cable Connections

USING A DIRECT CONNECT MODEM

Direct connect modems have a switch on them which allow them to be **on-line** or **local test**. When they are on-line they will hold the call and allow you to replace the receiver, thus allowing the communication to be uninterrupted by the noise in the room. There will be a lead with a telephone jack plug on the end which is plugged into a telephone socket. There is usually a socket on the back of the modem into which your telephone can be plugged. Modems with *auto-dial* don't need a telephone, and so don't have one of these sockets. Other budget modems may not have this socket either, but

you will need to use the telephone to dial the number of the service you want to reach, and to hear the connecting tone. In this case you will need to get a two-way adaptor which will allow you to connect two devices into the telephone socket, or have a double socket fitted.

Make sure that all the relevant settings of the modem and the communications software (see Chapter 11 if you are using the Amstrad communications program **mail232**) are correct for the computer that you are calling, and that they are switched on and running. With the modem switched to **local test**, pick up the telephone handpiece and dial the number of the distant computer. You may hear one or two ringing tones and then a continuous tone. Switch the modem to **on-line** and replace the handset. Press the return key a few times and the distant computer should respond with a message on your screen. Connecting up like this is known as **logging on**. When you **log off** at the end of the communication make sure that you switch the modem back to **local test**, otherwise it will hold the line with possibly disastrous consequences for your telephone bill! I always pick up the telephone to check that the line has been dropped.

USING AN ACOUSTIC COUPLER

Acoustic couplers are usually simpler than direct connect modems. Plug the coupler into the modem. They are often battery operated, but may need the battery charging. Switch on the Amstrad and load and run the communications software. Make sure that it's set to the right baud rate. Dial up the distant computer, and when you hear the reply tone, fit the telephone handpiece into the rubber cups. Make sure that they are connected the right way round. Press the <RETURN> key as with the direct modem. At the end of the communication, remove the handset from the coupler and replace on the telephone.

PACKET SWITCHSTREAM

If you are going to do a lot of data communications with host computers over long distances, either in the United Kingdom or overseas, it's worth knowing about **Packet SwitchStream**. This is a communications network operated by British Telecom, which is dedicated to data. It works in a different way to the PSTN, and so can be much more economic to use over long distances.

If you call a distant computer over the PSTN, you are connected through a circuit which is completely reserved to you throughout the duration of the call. The greater the distance, the more equipment you are occupying. The longer the duration of the call, the longer you are depriving other people of the use of it. So your bill is calculated on the distance of the call and the time.

Packet SwitchStream takes advantage of the fact that, with many data communications, the actual transmission and reception need not occupy the total time of the call. Therefore, even during the call, there is spare capacity on the network. So when data is sent to the network, it's split up into **packets** of, typically, 128 bytes each. Each packet has a header and footer (rather like you can have on a LocoScript page!). The header contains the address of the destination, and the footer contains sophisticated parity information to check for transmission errors.

The data is first sent to the nearest **packet switching exchange** or **pse**. In fact the whole network consists of these packet switching exchanges which are really powerful computers. The pse examines the address in the packet's header, and sends the packet over the network on high-speed, high-capacity transmission lines. Each pse in turn reads the address and sends the packet on until it reaches its destination. As each packet is completely independent, other packets from different senders and to different destinations can be interleaved on the network. All this is completely invisible to the user, to whom it appears exactly the same as a direct connection over the PSTN. But in this way the network is very efficient, and the charges are based largely on the actual amount of data sent rather than solely on the distance or total duration of the call.

Large (mainframe and the like) computers can be connected directly to Packet SwitchStream because they can assemble the data into packets and disassemble them again afterwards. Smaller ones (Amstrads for instance) can be connected to a **PAD (packet assembler/disassembler)**. There are public PADs at each packet switching exchange and these can be reached over the PSTN. This is known as **dial-up** access. There are pses in 22 centres around the United Kingdom, and 75% of all business users are within the local telephone call area of one of these. Another service connected to PSS, MultiStream, has local dial-up access in an even wider area. The snag for

dial-up users is that they cannot be *called up* over PSS, they can only originate calls. So it's most useful for accessing other services.

One of the greatest advantages of PSS is that the network supports many different standards and transmission speeds. Provided that your terminal can connect with Packet SwitchStream, then through PSS you can communicate with any other service on the network. This is regardless of, and without even knowing, which transmission rate the other party is using. Communication overseas is very easy, as PSS is connected to IPSS (international packet switching) allowing calls to 46 other countries around the world (this number is increasing every year).

Joining Packet SwitchStream

Your local British Telecom office will tell you where to apply to join Packet SwitchStream. For a fee you will receive a **network user identity (nui)**. This is a string of twelve characters and numbers. It allows the network to identify your account each time that you log on. This is your password to PSS; if anybody else knows it they can use the service and you will receive the bill, so guard it with care! For security, when you log on to PSS, only part of the nui is reflected onto your screen, so anybody watching won't be able to see what it is. You have to notify British Telecom as to how many different packet switching exchanges you will want to be able to log on to. You have to pay extra for each one, so you will probably want to use your local one, and others if you travel regularly to a particular place and want to use pss there.

Using Packet SwitchStream

When you join pss as a *dial-up user* you are given a list of telephone numbers for all the packet switching exchanges. All pses have several numbers, and you use the appropriate number for the transmission speed that you are using. Log on as described above, but when you hear the tone, instead of just pressing the <RETURN> key, you enter the code:

<RETURN><RETURN>A2<RETURN>

Enter this a few times until the pse responds with a message. You must then enter your **nui**. This has to be entered correctly, or the pse

won't recognise you. In the next chapter you are told how to program the computer to enter this for you at a single keystroke. If successful, the pse will ask for an **ADD**. This is the **network user address**. This is the identity of the service that you want to be connected to. If you enter this correctly, and if the service is able to accept your call, you are connected. From this point on, the communication will be the same as if the connection were made over the PSTN, except that there is less chance of line disturbance. When you log off at the end of the communication the pse will send you a **CLR PAD** message and show how many packets were used. Now switch the modem to **off line** in the normal way.

11 Using MAIL232

INTRODUCTION

There is a communications program supplied with the Amstrad. The program is called MAIL232, and it's on the LocoScript disc. You won't see it unless you list *hidden files* (<f8>=Options on the disc management menu page). You can buy other communications software for the Amstrad, but unless you will be doing a lot of communications and need something more sophisticated, or you need to access Viewdata services, then MAIL232 will do quite well to start off with. It won't work under LocoScript, it's a CP/M program. Of course, it won't work properly without the CPS8256 Centronics/serial adaptor. But even without it, you can run the program to see what it looks like.

MAIL232 will be easier to use if you copy it to a special disc, together with CP/M and some utilities. How to do this is explained later in this chapter. To see what MAIL232 looks like, load CP/M into the Amstrad and then put the LocoScript disc into the drive. Type:

mail232<RETURN>

and mail232 will be loaded and run.

Key <f1> allows you to change the terminal parameters. This is done in the same way as with the LocoScript menus; use the <↓> and <↑> keys to move the cursor down and up, and use the <+> key to change the values. Only the <+> key works, so if you go past the value that you want, you must go on round the loop until you come to it again. It's a slight nuisance that the program always loads at a setting which isn't very often used, but later in this chapter is a

way of programming the Amstrad so that pressing just one key will set the terminal parameters to the setting you usually use. Press <EXIT> to get rid of the menu, and the values set when <EXIT> is pressed are then effective.

Key <f3> allows you to send files which you have already prepared on disc (more about those later), or to save onto disc anything received. A pull-down menu allows you to enter the names of the files to be sent or received. The names can be entered in advance if you wish. You leave the menu by pressing <EXIT>, but to send or save, you must press <ENTER> with the cursor over the appropriate file name. When a file is received, it is saved initially in the Amstrad's internal memory. When all the information to be stored has been received, press <ALT>+<STOP> together, and that information is saved to the disc in the drive under the file-name previously entered. Be sure that you are pressing <ALT> *before* you press <STOP>, if not you will lose the communication and also the information!

Key <f5> provides a menu which allows you to toggle between **local** and **on-line**. Local shows on the screen anything entered at the keyboard. This can be useful to check the terminal before you connect to a modem. During transmission, information sent from the terminal is usually reflected by the host computer, and most modems reflect when they are in **local** test mode. However, if you are sending directly to another micro, then you will probably need to use **local** in order to see what you are sending.

Key <f7> allows you to leave mail232 and return to CP/M.

MAKING A COMMUNICATIONS DISC

The best way to use mail232 is to copy it onto a disc which you just use for comms. That way you can arrange the disc so that it automatically loads and runs mail232, and even programs some of the keys to do useful things.

First, make sure that you have an empty formatted disc ready, then put your CP/M disc into the Amstrad. *Boot up* CP/M, and then

using **PIP**, copy the following commands to copy some files into the memory disc:

```

pip m:=a:j14cpm3.ems<RETURN>           (your version
pip m:=a:pip.com<RETURN>                 number may be
pip m:=a:erase.com<RETURN>               different, check
pip m:=a:setkeys.com<RETURN>             with dir)
pip m:=a:submit.com<RETURN>
pip m:=a:basic.com<RETURN>
pip m:=a:rpel.bas<RETURN>

```

Once all these are copied, type in:

```
m:<RETURN>
```

and then at the prompt **M>** type:

```
dir<RETURN>
```

to check that they are there. Now put the LocoScript disc into the drive and type:

```
pip m:=a:mail232.com[r]<RETURN>
```

*([r] is used because **mail232** is in the **systems directory** which isn't usually searched)*

Check again in the **systems directory (dirs)** that this has been copied.

Put the new blank disc into the drive and type in:

```
pip a:=m:*.r]<RETURN>
```

This should copy all the files in the memory disc onto the new disc. You will be able to delete some of these again later, but you will need them all to set up your comms disc.

In **mail232**, most of the programmable keys of the Amstrad aren't used. Only **<f1>**, **<f3>**, **<f5>**, **<f7>**, **<↓>** and **<+>** are used.

Figure 11.1 is a list of the keys that can be programmed (those with an asterisk are used by **mail232** and re-programming them should be avoided in this case for obvious reasons).

There are several strings of characters that you will find that you will need frequently to enter into the Amstrad when you use it as a communications terminal. These include setting up the **terminal parameters** to match the settings on the modem and the system that you are communicating with. If you are using Packet SwitchStream, there is the **network user identity** and the **network user address**. If you are using a service such as Telecom Gold or a database there will be

Key	Expansion token	Normal string
<STOP>	#80	↑C
<f1> *	#81	↑Z
<f2>	#82	↑Z
<f3> *	#83	↑Q
<f4>	#84	↑Q
<f5> *	#85	↑S
<f6>	#86	↑S
<f7> *	#87	↑P
<f8>	#88	↑P
<DEL→>	#89	↑G
<←DEL>*	#8A	↑H
<CAN>	#8B	↑X
<CUT>	#8C	↑U
<PASTE>†	#8D	↑W
<FIND>	#8E	↑J
<EOL>	#8F	↑F↑B↑B
<LINE>	#90	↑F↑B
<↑>	#91	↑_
<+> *	#92	↑V
<←>	#93	↑A
<→> or <CHAR>	#94	↑F
<RELAY>	#95	↑R
<↓> *	#96	↑↑
<ALT>+<DEL→>	#97	↑K
<→>	#98	↑\
<ALT>+<↓>	#99	↑E
<ALT>+<←DEL>	#9A	↑X

* - used in mail232

† - useful in CP/M

(↑ represents *control*. This is obtained by pressing <EXTRA>+U)

Figure 11.1 Programmable Function Keys

your **identity**. You'll need to enter all these correctly if you are to get on to the service without being rejected. I don't recommend that you program passwords into the Amstrad, for security reasons these are best kept in your head.

TERMINAL PARAMETERS

When first loaded up, mail232 has the following terminal parameters set:

```
Tx Baud Rate . . . . 9600
Rx Baud Rate . . . . 9600
Data bits . . . . . 8
Parity . . . . . NONE
Stop Bits . . . . . 1
H/W Handshaking .. OFF
```

Typical data communications parameters would be:

	<u>V.21</u>	<u>V.23</u>
Tx Baud Rate	300	75
Rx Baud Rate	300	1200
Data bits	7	7
Parity	EVEN	EVEN
Stop Bits	1	1
H/W Handshaking ..	OFF	OFF

If you want to program a key to make the Amstrad change the terminal parameters, you must do it in the same way that *you* would do it by pressing all the necessary keys in turn. To find out how, type: **mail232<RETURN>**

to run the program and try it. Once into mail232, to change the parameters for 300/300, 7 bits, even parity, one stop bit and handshaking off you would need to press:

<fl>

— to call up the appropriate menu

<+><+><+><+><+><+><+>

— to change the Tx rate from 9600 to 300

<↓>

— to move the cursor down to Rx

<+><+><+><+><+><+><+>

— to change the Rx rate from 9600 to 300

<↓>

— to move the cursor down to Data Bits

<+><+><+>

— to change the Data Bits to 7

<↓>

— to move the cursor down to Parity

<+><+>

— to change the Parity to EVEN

<EXIT>

— to leave the menu and go back to mail232.

Similarly, to change the parameters to 1200/75 would need the following key depressions:

<f1><+><+><+><↓><+><+><+><+><+><+>

<+><+><+><↓><+><+><+><↓><+><+><EXIT>

Just seeing these listed out makes you realise how tedious it would be to have to do this every time that you want to use mail232, so why not let the computer do it for us?

To do this, we'll use the CP/M utility program *setkeys*. This reads a file on the disc and programs the function keys according to the information that it finds there. So if we want to reprogram a key that's not normally used in mail232, let's say the function key <f8>, to change the parameters to 300/300, the file should read:

E #88 "↑Z↑V↑V↑V↑V↑V↑V↑V↑V↑'30'↑V↑V↑V↑V↑V↑V↑V↑'30'
↑V↑V↑V↑'30'↑V↑V↑'27'"

To use the same key to change the parameters to 1200/75, the file would be:

E #88 "↑Z↑V↑V↑V↑V↑'30'↑V↑V↑V↑V↑V↑V↑V↑V↑'30'
↑V↑V↑V↑'30'↑V↑V↑'27'"

where E stands for *expansion string*

#88 is the <f8> key

↑Z gives the same effect as pressing <f1>

↑V gives the same effect as pressing <+>

↑'30' gives the same effect as pressing <↓>

↑'27' gives the same effect as pressing <EXIT>

Now in order for it to work, we must get these key definitions into a file on the disc. This is most easily done using the BASIC text editor called **rpedit**. This is one of the programs that we copied onto the comms disc. If you're still in mail232, press <f7>, followed by <↓> and <ENTER> to get out into CP/M. Then key in:

basic rpedit <RETURN>

This will load **basic** and **rpedit**, and then *run* rpedit. When the main rpedit menu page comes up, press <f3> to edit a new file. Give it the name **params**. After you press <ENTER>, the screen will change to the editing screen. Now type in the required instruction line E #88 "↑Z ..., as shown above. Remember that to get ↑ you press <EXTRA>+U.

When you have finished the line and satisfied yourself that it's correct, press <EXIT> to save the file to the disc.

Now, still in the rpedit menu page, press <f3> again to create another new file. Call this one **profile.sub**. Note that rpedit is rather inconsistent here with the other programs supplied with the Amstrad; in order to get the cursor to the second part of the file-name (the file-name extension), you must press either the **full stop** (.) or <→>. The contents of this file should be:

**setkeys params
mail232**

When you have finished, press <EXIT> twice, once to save the file and the other to leave rpedit. Now reset the computer by pressing <SHIFT>+<EXTRA>+<EXIT>. The Amstrad should load CP/M, load setkeys and params, and then load and run mail232. Once into mail232, press <f1> to see the settings of the parameters. Then <EXIT> and press <f8> (that is - <SHIFT>+<f7/f8>). The parameters menu will flash on, the parameters will change very quickly and the menu will disappear again. Press <f1> again to check that they are set as they should be. If they are wrong, the **params** file can be altered to correct them. Remember though that the parameters have been changed by simulating a number of key depressions from a given starting point. If you press <f8> for a second time, the same sequence of changes will be effected giving a completely different and probably quite useless result. The parameters can still be changed in the normal way using <f1>, <↓> and <EXIT>. When you use mail232 for normal communications you needn't check the

parameters before you press <f8>, nor afterwards unless you have reason to doubt that **setkeys** was loaded.

You can program more than just one key to make mail232 easier to use. When you are using data comms there are often codes that you need to enter into the terminal in order to get access to services. I use my Amstrad to access Telecom Gold over Packet Switch-Stream (see Chapter 13 for more information on these). This requires three long codes to be entered, the PSS **network user identity** (**nui** – 13 characters), the **network user address** (**nua** – 10 characters) and my Telecom Gold **identity** (**ID** – 9 characters). I have these three entered into the function keys <f2>, <f4>, and <f6> (see below), so that by just pressing the appropriate key the code is entered correctly. I don't have my password programmed into the Amstrad, for security reasons it's much better to remember this and enter it by hand.

E #82 "NTLGOLD901TYM↑M" *key* <f2>

E #84 "A219201004↑M" *key* <f4>

E #86 "ID ABC123↑M" *key* <f6>

These lines are simply added to the **params** file.

Of course you can choose what strings you program into which keys. There is a limit to the number of characters that you can enter before the appropriate part of the Amstrad's memory is full. I've put in about 80 extra characters before it starts to ignore lines. You should also think carefully about which keys you program. If you want to go out of mail232 and use CP/M, the key definitions will stay put. So if you re-program keys that are used in CP/M (some of the cursor keys for instance), you may have problems. The easiest way to reset the keys is to re-load CP/M again from the start. The function keys aren't usually used in CP/M so they are safe to change.

When you have finished creating the necessary files, you may wish to delete all the unwanted files from your comms disc. Those that will be needed are:

J14CPM3.EMS (CP/M)
 PIP.COM
 ERASE.COM
 SETKEYS.COM
 SUBMIT.COM
 PROFILE.SUB

PARAMS
and **MAIL232.COM** (in the systems directory)

You can now make all these files *systems* files so that they don't normally show when you type **dir**. This makes it easier to see what transmission and reception files there are on the disc. You can also set them to **read only**, so that you can erase any text files on the disc without danger of wiping the programs. To do this you can use **set**. Copy **set** from your CP/M disc into the memory disc, select the memory drive and type:

```
set a:*. * [sys]<RETURN>
```

```
set a:*. * [ro]<RETURN>
```

You will still be able to list these files using **dirsys**.

MAKING ASCII FILES

Key <f3> allows you to send and receive files. They must be ASCII files, so you can't send LocoScript files directly. This is because LocoScript uses code numbers well outside the ASCII range, and also contains a lot of control codes. Instead you can create ASCII files from LocoScript (version **1.2 only**), but you must do this in advance, before you go into mail232. To do this, go into LocoScript, and in the disc management menu place the cursor over the file that you want to send. Copy it into the memory disc using <f3>, and then place your comms disc (or the disc that you will be using during communications) into the drive. Press <f1> for disc change. Now with the cursor over the file in the memory disc press <f7>=
Options. One of the options you are presented with will be **Create an ASCII file**. Choose this and you will be asked to choose the destination disc and group. Choose the disc in the drive. Call the new file by an easily remembered name, because it's not possible to get the disc directory during mail232. Then you will get a second pull-down menu, at the bottom of which you have the choice of **simple text file** or **page image file**. A simple text file will have no formatting except for line returns. Line lengths, tabs, etc will be ignored. If you are sending the text between word processors, this can be useful, because the destination machine can then format the text with its own word processing program. The main problem is that if the line length is greater than 80 characters, it disappears off the right of the mail232 screen and you can't see it being transmitted. It does get sent, but some mailbox and other services won't accept a long line

length, so for most purposes it's probably safer to use a **page image file**. If you are sending text to an external printer, then you need a page image file for this too.

When a page image file is created all the formatting will be interpreted, and converted into spaces. The text will *look* the same as under LocoScript except that it cannot be so easily re-formatted. Remember also that if you are going to transmit that file, it is likely to be much longer than the original because it now has a lot of spaces in it, all of which need to be transmitted. If you are going to make a page image file, it's well worth while editing the header of the LocoScript file before you start to move the left margin to 0, so that there are no leading spaces at the beginning of each line.

Some people who do a lot of text transmission find the whole business of using LocoScript for making ASCII files rather tedious as it means leaving LocoScript and going into CP/M to transmit and *vice versa*. They use a CP/M word processing program such as New Word or WordStar which produces ASCII files as normal files. They tell me that it's much more convenient that way, but as I prefer LocoScript, I'm quite prepared to put up with the modest inconvenience.

TRANSMITTING A FILE

To send a file from mail232, make sure that you have the required ASCII file on a disc in the drive, or recorded in the memory disc. Press <f3> and there is space on the pull-down menu for you to enter the name. You don't need to send the file immediately. If you press <EXIT> you will lose the menu, but the file won't be sent. The file-name remains on the menu however until you press <f3> again and press <ENTER>.

RECEIVING A FILE

<f3> also allows you to record received text onto a disc. Enter the file-name under which you want the text saved. Again, the recording only starts if <ENTER> is pressed. While the text is being received it's stored in the memory disc. It is saved to the floppy disc at the end of reception, when <ALT>+<STOP> are pressed. **Caution** – make sure that you have pressed <ALT> *before* you press <STOP>. If not

you are likely to find yourself back in CP/M having lost the communication. If you are receiving a file you must make sure that there is enough room for it on the disc. Mail232 doesn't give an error message if there isn't, but won't save the file.

USING MAIL232 WITH TWO DISC DRIVES FITTED

The spaces for file-names under <f3> don't allow you to specify which disc drive they are on, therefore they must be transmitted and received from the currently selected drive (usually drive A:). If a second disc drive is fitted it's most convenient to create an ASCII file for transmission onto your *communications disc*. In LocoScript, place the LocoScript file disc in drive B:, and the communications disc in drive A:. Using <f7> in the *disc management menu*, create an ASCII file onto the *communications disc* in drive A:. When inserting a received ASCII file into LocoScript, place the *communications disc* in B: if the transfer is to a CF2 disc, or in A: if the transfer is to a CF2-D disc.

USING AN ASCII FILE IN LOCOSCRIPT

If you try to edit an ASCII file in LocoScript you will receive the error message **Not a LocoScript file**. This is because LocoScript cannot find a *file header*. The solution is simple. Create a new document in a group with an appropriate template. Once the document is open, press <f7> and choose **Insert text**. The ASCII file can then be inserted into the new document which can be edited in the usual way.

UNFORMATTING A DOCUMENT

If the document that you received and inserted into LocoScript is a *page image file* and contains lots of spaces and line returns, it could be rather tedious to edit. Fortunately there is an easy way to do it, using the <EXCHange> key.

The biggest problem is to take out all the line returns, but retain the paragraphs. So first you must identify the paragraph ends. Press the <EXCH> key and in the **find** space print two <RETURN>s (↵↵). Then press the <↓> key and in the **exchange** space print a character that isn't used in the document (a § for example). Now with the <↓>

key move the cursor down to **Automatic exchange to end of doc**, and press <ENTER>. The Amstrad will now replace all the paragraph endings with the symbol that you chose.

Now to get rid of all the unwanted line returns. Move the cursor back up to the top of the document. Press the <EXCH> key again and this time delete one of the returns which are still in the find space. Delete the symbol in the exchange space and replace it with a single space. This is necessary because if you simply removed all the returns, the words at the end of each line and the beginning of the next would run together. Move the cursor to the bottom of the menu and press <ENTER>. Now all the returns have been removed, but all the words are separate.

To put the paragraphs back in, move the cursor to the top of the document and this time replace the paragraph symbol with two <RETURN>s. Now the document can be edited as a LocoScript file.

ALTERNATIVE COMMUNICATIONS SOFTWARE

MAIL232 has the great advantage that it's free. It's quite adequate for bulletin boards and simple electronic mail, although it can't be used to access Viewdata services. Any other communications program is likely to cost at least £30. However if you are going to do a lot of communications, then you might find that the additional outlay pays off in the end in terms of additional facilities (the ability to use discs in other drives and see the disc directories for instance) and convenience. Chapter 9 mentions an alternative software package which has these features.

12 Direct Communications

INTRODUCTION

You can use your Amstrad word processor to send and receive files from other computers which are close by. There's not much point doing this with another Amstrad, because you could just take the disc out of one and put it in the other, but it's very useful where the disc formats are different. Files are most easily sent in ASCII code which is standard between most computers. As the Amstrad is a CP/M computer, there is a wide range of software available under this operating system. Locomotive BASIC is very similar to Microsoft BASIC, so these programs might be interchangeable. Transferring electronically is much easier than re-keying.

To do this you will need the **CPS8256 Centronics parallel/RS232 serial interface**, and a suitable cable. The communication will be through the serial port, that is the lower of the two outlets on the adaptor. There is more than one way that data can be transferred between the Amstrad and another computer, but the simplest is probably using the mail232 program supplied. Another way would be to use the pip utility under CP/M, and this is described in the CP/M manual.

In this chapter I will describe how you can transfer text between the Amstrad and three other popular types of equipment, a **BBC micro**, a **Tandy** lap portable computer and a **Brother** portable electronic typewriter.

To transfer text using the **MAIL232** program you must prepare ASCII files. This is easily done from LocoScript, and is described in Chapter 11. If you are transferring to another word processor you

must remember that LocoScript is able to use a very wide range of characters that are not available to others, and so these will not be sent. There is a list of the characters which *can* be sent in Appendix 2. Also many of the special effects like different letter pitches, bold, italic and underlined cannot be transferred either. But transferring text is still a lot easier and less prone to typing errors than re-keying the text.

COMMUNICATING BETWEEN THE AMSTRAD AND A BBC MICRO

One of the most popular home computers for word processing before the Amstrad word processor came along was the BBC micro. There are several word processing packages for this, and it has the advantage of a built-in serial port as standard. It may be that you already have word processing files on a BBC, or that you know someone who has, or else you may want to transfer files from your Amstrad to a BBC. This is quite easy provided that you have a CPS8256 serial adaptor for the Amstrad. You don't even need to have a communications program for the BBC, you can feed the text directly into a word processing program like **Wordwise**. Communication between the Amstrad and a BBC micro is illustrated in Figure 12.1.

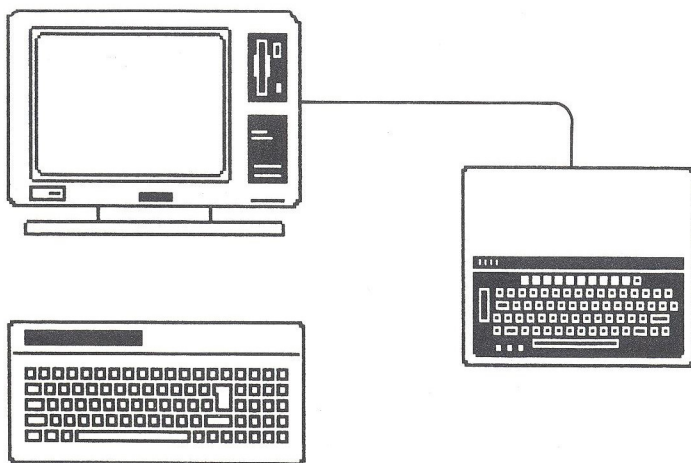


Figure 12.1 Amstrad to BBC Communications

The serial port on the BBC micro conforms more or less to RS423. This is a more precise standard than RS232 because the voltages allowed are more precisely specified. However, the socket used on the BBC has only five pins (a DIN 'domino') and it is possible to insert the plug in two ways (the right way and the wrong way!). Inserting it the wrong way seems to do no harm, except that the communication won't work. Suitable leads can be bought, although it may be easier (and cheaper) to find the components and make them up yourself. Figure 12.2 shows you how to connect up a lead to go between a BBC and the standard 25-way D plug on the Amstrad. All connections are shown as you would see the plugs as you are soldering them. You will need to use cable with at least five wires. Short lengths don't need to be electrically shielded, but for anything over a metre it is recommended that four core screened cable is used, the metal screen can be used for the earth connection.

Care should be taken when connecting the domino plug into the BBC to make sure that it's the right way up. There is usually a gap in the metal shielding which fits into the socket, and using these wiring schemes, the correct position for this is *upwards*.

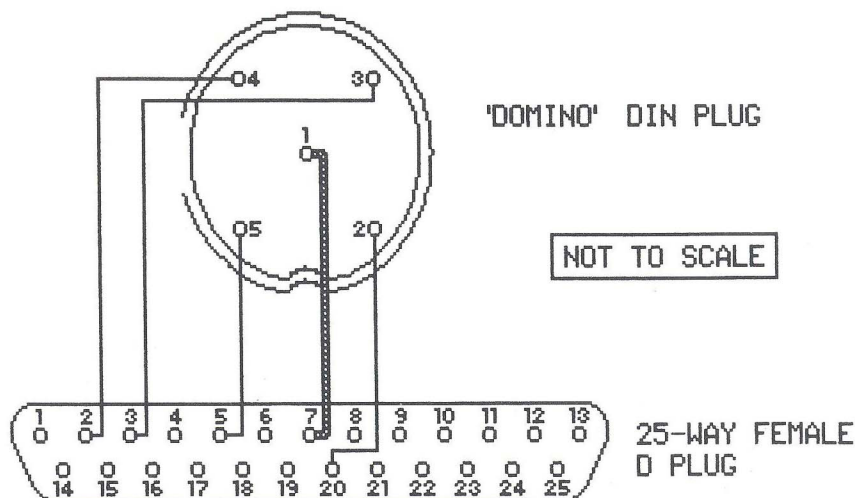


Figure 12.2 Amstrad to BBC Cable Connections

Transferring Text from BBC to Amstrad

In order to do this you must have an ASCII file prepared (either using the ***SPOOL** command of the BBC, or use **option 8** in Wordwise). Place this in the BBC's disc drive. To transfer from the BBC you don't need to be in Wordwise. Set up the two computers to the same baud rate. Transfer seems to work well up to 1200 baud. Load up mail232 on the Amstrad and set the terminal parameters (using <f1>) thus:

```
Tx Baud Rate . . . . 1200
Rx Baud Rate  . . . 1200
Data bits    . . . . . 8
Parity       . . . . . NONE
Stop Bits    . . . . . 2
H/W Handshaking . ON
```

Then press <f3> and set the Amstrad to receive using a suitable file-name and press <ENTER>. The Amstrad will now wait to receive. Set the baud rate on the BBC. As you only want to transmit enter:

```
*FX8,4 <RETURN>
(this sets the transmit baud rate to 1200)
```

Then to send the text through the RS423 port enter:

```
*FX3,5 <RETURN>
(this sends output to screen and serial port)
```

Now type:

```
*TYPE {file-name}
```

The transmission will now take place. When it is finished press <ALT>+<STOP> on the Amstrad to save the file to the disc.

Transfer from Amstrad to BBC

In this direction it's most convenient to send the text directly into the *Wordwise* program. Prepare an ASCII file on the Amstrad, and load mail232. Set the terminal parameters as shown above. Now prepare the BBC. Load Wordwise, and in the menu screen enter:

```
*FX7,4 <RETURN>
(this sets the receive baud rate to 1200), and
*FX2,1 <RETURN>
(this sets the RS423 port as the input device)
```

It's important to enter the ***FX2,1** last, as this will disable the keyboard. Fortunately the <ESCAPE> and <BREAK> keys still work, these are the only ones we need. Press <ESCAPE> twice and this

will bring you to the editing screen. Try to type something if you like, but the keyboard is out of action. Now set the Amstrad to transmit the file with <f3>. You won't see anything on the Amstrad's screen, but it will appear as if someone is typing very quickly into Wordwise. When the transfer is complete, press the <BREAK> key on the BBC, it'll send you back to the Wordwise *menu page*, and restore the BBC's keyboard. Now your file is in Wordwise and can be edited or saved to disc.

COMMUNICATING BETWEEN THE AMSTRAD AND A LAP PORTABLE COMPUTER

Lap portable computers are becoming very popular with people on the move. These are completely self-contained and battery operated, so they can be used anywhere. In fact they can be used resting on your lap, which is how they got their name. The screen is usually a liquid crystal display of limited size, and although these computers usually come with a text editing program adequate enough for typing in, they are not really up to word processing that text afterwards. It's an easy matter to transfer that text into your Amstrad, for editing, formatting and printing out. You can also transfer Amstrad files to a portable.

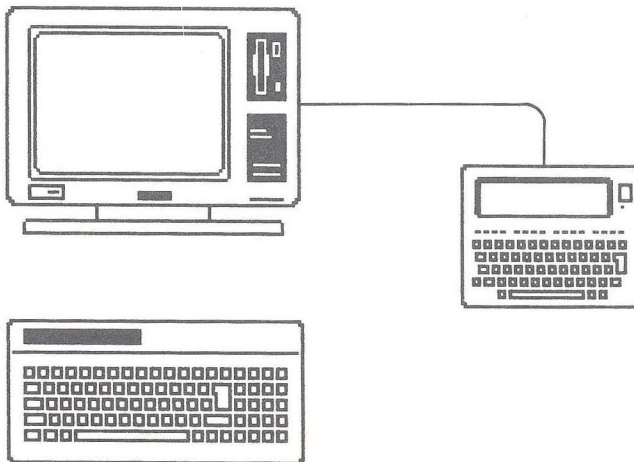


Figure 12.3 Amstrad to Tandy Communications

The following is a description of how to transfer files between an Amstrad and a **Tandy TRS-80 model 100**, one of the most popular models. This is very similar to several other makes. It comes with a built-in text editor and a communications program. Amstrad to Tandy communication is illustrated in Figure 12.3.

All that you will need apart from the CPS8256 serial adaptor is a suitable lead. The RS232 port on the back of the Tandy is a 25-way D socket, similar to the one on the Amstrad adaptor, except that it is a *female* socket. So the lead that you need will have a 25-way male D plug on one end and a 25-way female D socket on the other. Figure 12.4 shows how you should wire it if you make one up yourself. Only 5 core cable is needed, 4 cores with a screen will do. This type of cable is known as a **null modem** cable. It will also work with a serial printer (if the plugs are compatible), *but is different from the one needed to work with a modem*.

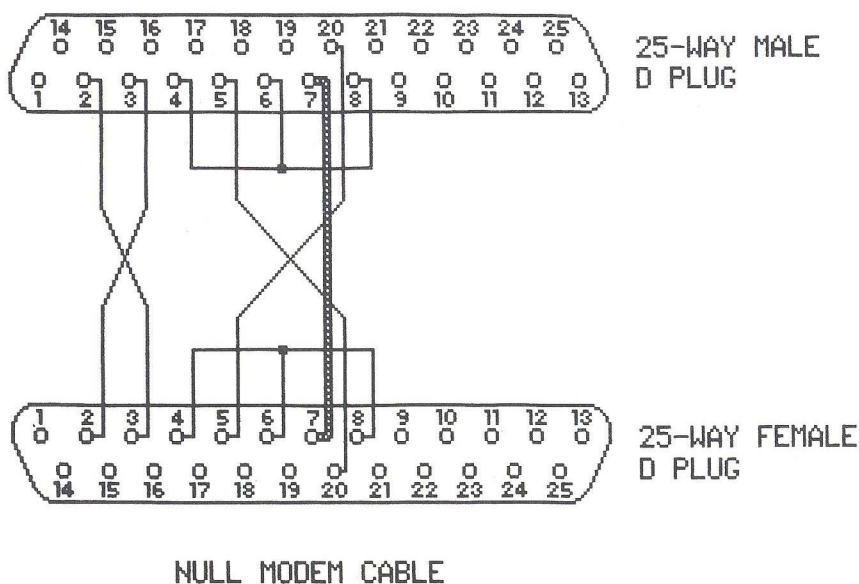


Figure 12.4 RS232 Null Modem Cable Connections

Setting up the Two Computers

Set up the two computers and plug in the RS232 lead. Load up mail232 in the Amstrad. When you switch on the Tandy, you are shown a menu of the programs available, and a directory of the files held in the computer's memory. The memory has a battery back-up which keeps all the files safe until they are either deleted, the computer is reset or the batteries run down completely. There is even a supplementary back-up which keeps the files safe while you change the main batteries. Using the cursor keys, select the TELCOM program. When the screen changes you will see the terminal parameters that are set. The *default* settings are 300/300 baud, 7 bits, **parity ignored** and **hardware handshaking enabled**. The main instruction manual will tell you how to change these if you want to. The important thing is to make sure that the Amstrad is set to the same parameters.

Transferring Text from the Tandy to the Amstrad

First of all, prepare the Amstrad. Make sure that there will be room on your disc for the file to be transferred. Then press <f3> and move the menu cursor to the receive line and enter the name under which you want to store the file. Press <ENTER> and the Amstrad will now wait for the transfer to take place.

The Tandy menu page will show you the names of the files in the memory that can be transferred. Like the Amstrad, you can't see these when you are in terminal mode, so copy them down somewhere if you can't remember them. The text (or **DO**cument) files will have a file-name extension of **.DO**, but you don't need to enter this unless you want to erase the file. Once you have entered the **TELCOM** program, the function keys are labelled along the bottom line of the readout. There are two screens in **TELCOM**, the first one is for setting the terminal parameters, and the other is for the communication. To get to the second screen, press <f4> **Term**. To transfer text **from** the Tandy you want to use the **U**pload function which is <f3>. You will be asked:

File to upload?

Enter the name of the file to be transferred. Then you will be asked:
Width?

This is the number of characters to be sent before a line return. As the Amstrad has a 90 character wide screen, enter that number. When you press <ENTER>, the transfer will start, but you can only see it on the Amstrad's screen. Even so there won't be any line feeds, so the text will keep over-printing itself on the bottom line of the screen. Don't worry, it's being saved. When you come to the end of the file (it may be a good idea to always put an end of file message on the last line of files to be transferred), press <ALT>+<STOP> on the Amstrad so that the file is saved to disc and the file closed. The Tandy will have automatically gone back to normal terminal mode at the end of the file.

You now have an ASCII file on your Amstrad disc; this can be used directly with a CP/M word processing package, or inserted into a LocoScript document (see page 75).

Transferring Text from the Amstrad to the Tandy

You may have a document in the Amstrad that you want to take with you in the Tandy, perhaps an address list, or a product list for reference. Follow the instructions above for setting up the two computers. Then prepare the Tandy. Go into the **TELCOM** program, and press <f2> for **download**. You will get the question:

File to download?

Enter the file-name under which you want to store the file and press <ENTER>. The Tandy will now wait to receive the file.

With the Amstrad in mail232, and the terminal characteristics matching those of the Tandy, press <f3>, enter the name of the file to be sent and press <ENTER>. The file will now be transferred, but you will only see it on the screen of the Tandy. Where a line runs over the edge of the screen, there will be a line feed and the text will scroll up, but where there is a <RETURN> embedded in the text, the next line will be printed over the former one. This does not stop the file from being recorded correctly. You can check that the file is alright at the end of the transfer by leaving the telcom program by pressing <f8> twice. This will take you back to the Tandy's main menu and directory. You should see your file-name in the directory. Enter the **text** program and call up the file, and you will see it as it has been saved.

COMMUNICATING BETWEEN THE AMSTRAD AND A PORTABLE ELECTRONIC TYPEWRITER

Portable electronic typewriters have become very popular recently. Relatively cheap, they are portable, usually working off batteries, and with a limited memory, they allow some text editing. They can also give quite good print quality and are very quiet when printing out. Some models have a built-in RS232 connector, and many people have bought them to use as portable electronic mail terminals. If you have one of these machines you may wish to transfer text between the two. This could be because you have entered text into the typewriter away from home and want to edit it properly, or want to incorporate it into another document. It could be that if your Amstrad printer fails then you want to use the typewriter as a temporary printer. The following instructions concern how to connect the Amstrad with a Brother EP44 electronic typewriter, which was one of the simplest and least expensive models with a memory and an RS232 port. This model only has 2 kbytes of memory, and text cannot be transferred into the memory through the serial port, only out.

Setting up the Two Computers

Connect the two devices together using a **null modem** cable exactly as for the Tandy (see Figure 12.4). Transfer to and from the Amstrad is most easily effected through **mail232**. The settings are:

```
Receive . . . . . 300
Transmit . . . . . 300
Data bits . . . . . 8
Parity . . . . . NONE
Stop Bits . . . . . 2
H/W Handshaking . ON
```

The parameters on the Brother are as follows:

```
Baud rate . . . . . 300
Bit length . . . . . 8
Parity . . . . . N
New line . . . . . CR + LF
Code . . . . . 7 bit
ER setting . . . . . Y
```

Note the setting for **new line** – a carriage return and a line feed. This

has two effects, when sending **from** the Brother to the Amstrad, it makes sure that the lines on the Amstrad aren't printed on top of each other. When using the Brother as a printer for the Amstrad, single line spacing is possible.

Transferring from the Brother to the Amstrad

Load and run **mail232** in the Amstrad, and set the parameters as above. Press <f3>, enter a file-name in receive and press <ENTER>. The Amstrad will now wait to receive some text. To send the text from the Brother, press <TEXT>. When the whole file has been transferred, press <ALT>+<SHIFT>. The new file will now be saved to the Amstrad's disc.

Transferring from the Amstrad to the Brother

The Brother EP44 won't accept text into its memory through the RS232, you can only type it in through the keyboard. Other models may allow you to do this, so look in the instruction manual. However, the EP44 *can* be used as a second printer. To do this you simply prepare an ASCII page image file of the document that you want to print, and send it to the Brother using mail232, using <f3> **Transmit**. Make sure that there is paper in the Brother before you press <ENTER>, as printing will start straight away. You can use the same settings as above, but if you do so, every 4 lines or so you will get an error message on the Amstrad:

SIO not ready, Retry, Ignore or Cancel?

You will need to press the <R> key, whereupon printing will resume. This is a bit tedious. However, if you turn handshaking off, then you will start to print out rubbish after 4 lines. This is because at 300 baud, the Amstrad is sending out text faster than the Brother can handle. The most simple solution is to send the text more slowly. The next speed down is 110 baud, and the Brother can handle this very well. The same baud rate must be set on both machines.

13 On-Line Services

BULLETIN BOARDS

Bulletin boards are becoming increasingly popular with people who wish to use their computers to communicate with others. Bulletin boards are rather like public notice boards where anyone can pin up a message. You can dial one up, and when you are logged on, you can read the messages that are there, and if you wish you can leave one yourself. Some bulletin boards allow you to leave a message that can only be accessed with a password, so that confidential as well as public messages can be left, and so it can be used for electronic mail. They are run by organisations, or even by individuals. They consist of a computer and a modem (usually one that has an *auto-answer* board built into it) and special software so that it will work on its own without supervision.

Some bulletin boards are used by people with a special interest. In the United States, where bulletin boards have been going for longer than in the United Kingdom, there are over a thousand. Bulletin boards exist for such interests as cookery, needlework, weaving and religion. So people with these interests can call up the appropriate bulletin board to find out what is happening and leave messages of interest to others.

Calling up a Board

To call up a board with your Amstrad you need a serial interface, a modem and a telephone. The most popular standard for bulletin boards is **V.21**, that is a baud rate of **300/300**, and so if you are likely to be interested in this aspect of data comms you should note that

the simplest types of modem intended for use with **Viewdata** (Prestel and Micronet) only handle **V.23**, that is **1200/75** baud, and so are unsuitable.

If your modem is *multi-standard*, set it to the appropriate standard and to the **local test** setting. Then set mail232 to the correct protocol. Dial up the bulletin board's telephone number. It's best to look for one in your area to start off with, the bulletin board may be free, but you'll have to pay the telephone bill, and this can mount up alarmingly if you get carried away!

Most bulletin boards only have one telephone line in, and as most callers will be connected for several minutes, if the board is popular, you must expect to have to call a few times before you get a connection. For this reason, most bulletin boards have a time limit for each connection, after which the bulletin board will automatically disconnect. If you do get disconnected, it may not do you much good to try calling again immediately. You probably won't get a reply. The computer has to do some housekeeping, tidying up its files and so on, so it won't reply to any call for a minute or so after the last one has ended.

When you do finally get through you will hear the **carrier** tone. Put the telephone handpiece onto the acoustic coupler or else switch your direct connect modem from **local test** to **on line**. Then you may need to press the return key a few times until you get a response from the distant computer.

If you get no response, or if you just get garbage on the screen, then you should check that the terminal parameters that you have selected are the correct ones for the *bb* in question. If you get something that looks correct but with a lot of errors, then you probably have a very bad connection. If you are calling over a long distance, then you could disconnect and try again in the hope that you will get a better line next time. This problem will improve in time as telephone trunk routes are being replaced with digital circuits.

The *bb* will probably ask you to identify yourself and then you will be presented with a menu.

Ring-back

Some *bbs* operate on a system called **ring-back**. This allows the

board's telephone line to be used for incoming voice messages too. If you want to call such a board, dial the number and let it ring just once. Then hang up and dial again. This time you will be connected to the modem. If you let the telephone ring on the first time that you call, if there's someone there they'll answer it.

Finding out about BBs

Several computer magazines have regular features containing up to date numbers. These magazines include **Tele-Link**, **Personal Computer World** and **Educational Computing**. *Bbs* themselves are often very good sources of up to date numbers of other *bbs*.

One bulletin board that will be of particular interest to Amstrad word processor users was due to go on-line about the time that this book is published. It's being organised by the **Amstrad User Software Database**, a non-profit making organisation set up and run by Amstrad computer users. To find out more about it, contact AUSD, PO Box 11, Gosforth, Newcastle upon Tyne NE3 1RP.

MAILBOX SERVICES

There is a commercial equivalent of bulletin boards which is becoming very useful and important to many people for their work. These are **electronic mail** services. The concept is simple. Imagine a set of pigeon holes, let's call them mailboxes, each one designated to an individual user. Messages can be left for someone in their mailbox until they collect it. But with an ordinary mailbox you have to go there physically in order to inspect your mail, and that's where the convenience of the system starts to fall down. However, with electronic mail the messages aren't left in pigeon holes, but in the memory store of a large computer. The messages are put there from computer terminals. Through the power of communications, the messages can be sent from anywhere that there's a telephone, and just as importantly, they can be collected from almost anywhere in the same way. Mailbox users can access their mailbox at any time, from anywhere that there's a telephone and a terminal. The terminal could be an Amstrad, or it could be a portable battery-operated micro, using an acoustic coupler and a public telephone. There are some types of people for whom the advantages of electronic mail are obvious. They are the sort of people who work away from base a lot.

They are almost impossible to get on the telephone, but electronic mail messages can be left for them in the knowledge that they'll get them the next time that they look in their mailbox, either when they get back, or when they call in from where they happen to be.

There are advantages for the recipients too. When they receive messages they can either deal with them straight away if they wish, or else when it's more convenient. The message doesn't disappear from the mailbox once it's read, only when it's been deliberately deleted.

GETTING A MAILBOX

There are several mailbox services available. The best known one in the UK is **Telecom Gold**. This is British Telecom's DIALCOM electronic mail system. It's a licensee of, and uses software developed by, Dialcom in the United States. Telecom Gold has several mainframe computers to store their users' mail; different computers are known as different systems. Thus someone may have her mailbox on System 81, while another has his on System 84. Users on one system can send messages to anyone on any other system. They can also send messages to people on Dialcom, an electronic mail service in the United States. It's not possible however to send messages between electronic mail services in the UK. Telecom Gold's computers are situated around London, which means that users in that area can dial directly over the normal public telephone network. Users further away are better off with the packet switched data network (see Chapter 10). However, they don't need a separate account with PSS, Telecom Gold have a public *network user identity* which subscribers to Gold can use. The PSS charges get added to the mailbox bill of course, but for those with only a small requirement for data transmission it's more economical.

Subscription charges for Telecom Gold tend to be geared towards commercial users, and indeed companies with a requirement for more than one mailbox. There are other ways of getting a mailbox on the same service, and one is to rent one through one of the mailbox retailers, that is, organisations who rent boxes in bulk, and sublet them to individuals. **MicroLink** is run by **Database Publications Ltd**, the publishers of the magazines **Tele-Link** (which is devoted to data communications) and **The Amstrad User**. There are other ser-

vices for more specialised users which provide mailboxes on Telecom Gold. These include **Textnet** (see later) and a service dedicated to the legal profession.

Another electronic mail service available in the UK is **One-to-One**. This offers similar facilities to Telecom Gold. An interesting, but not cheap, service available from One-to-One is that they will print out and deliver letters by courier in some major centres within two hours.

Modem manufacturers, and communications software suppliers, sometimes offer free initial subscriptions to mailbox services when you buy their goods. These offers have included Telecom Gold, One-to-One and Micronet from time to time. So if you're going to buy a modem or software, look around to see what's on offer.

Many large companies operate their own electronic mail services using the company's own computer. This can make economic sense if the facilities are available, companies also like the security aspects of this sort of set-up. However, contact with electronic mail users outside the company may not be so easy.

OTHER USES FOR ELECTRONIC MAIL

In fact many people find other uses for their mailboxes than just sending simple messages to other mailbox holders. Even if the terminal population is growing very quickly, for a long time to come there will still be more people *without* them than with. How can you use data comms to contact all these more quickly?

It's possible to send **telexes** directly from your Amstrad through a mailbox service like Telecom Gold or One-to-One. This saves having to have a dedicated telex line and terminal. You can also receive telexes from other people if they quote your mailbox number. There is a service which receives your incoming telex and leaves it in your mailbox.

If you subscribe to MicroLink you can send **telemessages** through your mailbox. Telemessages, a service of British Telecom, are the modern equivalent of the telegram. Telemessages sent before 10pm are delivered by first class post the next morning, anywhere in the UK or the USA. Of course you could dictate your Telemessage over the telephone for delivery, but sending it by electronic mail is

actually cheaper. Other services exclusive to MicroLink are Flora-Link, allowing you to order flowers by electronic mail, and British Rail Link allowing you to study British Rail timetables and order tickets.

TEXTNET

There are many people who use data communications to send more than just messages quickly over long distances. For many people, text is a perishable commodity, useless if delayed. Journalists are an obvious example, and so are translators. Much translation is done for commercial organisations with short deadlines. If text has to be sent by post over long distances to reach a translator with the right specialist knowledge, or simply one who is able to do the work at short notice, then much of the time available can be lost. Using data communications it's possible to send text anywhere in the world, have it translated and receive it back in the same day. An added advantage is that the text received back is already in the form of data, and doesn't need to be re-keyed, nor further proofread.

The Amstrad has a very full character set, and so is ideal for translators. However many of these characters are outside the range of other word processors and communications systems. The answer to this problem is to substitute unused characters within the ASCII range for special foreign characters. The **find and exchange** facility in LocoScript *can* be used for this, but a more efficient method is to use a program called **ROSETTA** (marketed by a company which enjoys the wonderful title of **Efficient Chips**). This will go through a file and automatically substitute characters according to a look-up table which the user can create and change.

Journalists and translators are rather exotic examples. Other services are springing up to take advantage of communications. There are transcription services which will accept manuscripts or audio-cassettes, type them up and then return the results to the customer's mailbox. In the opposite way they can accept text through the mailbox, a letter and mailing list perhaps, which they will then print out on a high quality printer and post.

Some typesetters and printers will now accept text sent through electronic mail. There are also **laser printing services** which will print out text files to a very high standard at reasonable cost. As well as

saving time, the communications link overcomes any problems of incompatible equipment. Where before text couldn't be transferred because of different floppy disc formats, data communication now forms the link.

Textnet is an organisation which brings together many different text related services. They offer mailboxes (on Telecom Gold) and maintain a database of service providers. The database is available to any Telecom Gold user for a supplementary charge depending on the length of time for which it's used.

DATABASES

There are many database services available, and these offer a vast range of information. Most of this is aimed at the commercial and industrial areas, and are available on subscription. Some news databases hold all the text of some newspapers. The Washington Post is available on-line *before* the paper version hits the streets. With so much data on-line, finding what you want can be time-consuming and expensive. However there is a retrieval system by which you enter key words. Supposing that you wanted to see any reports of microcomputers used for word processing. You would enter the two key words:

microcomputer wordprocessing

Special cross-referencing techniques will allow the computer holding the database to search through many millions of words in a few seconds. In the example shown you would probably receive a reply like this:

152,256 occurrences

This is a polite way of telling you that you should be more precise with your search. So now you might add an extra word to search on, say:

Amstrad

So now the computer will look for all the reports that contain those three words, and the number of references will be fewer. Eventually you would be able to narrow the search down to a manageable number of references. Then you can either read those on the screen, or download them onto a disc, or even have the articles printed out by the computer and sent to you.

Many libraries, public and specialist, now offer database search services. Many databases are in other countries, and access is most

economical using PSS. The Amstrad makes an excellent terminal for this purpose.

VIEWDATA

Viewdata is a communications standard which was originally intended to be used on modified television sets. The best known example of a viewdata service is Prestel, operated by British Telecom. The system has proved very popular with information services where the terminal is used by the public, in particular in travel agents. There are also other services such as home banking, where you can access your bank account to pay bills from your home (or wherever you happen to be). Some public authorities have set up viewdata information services for the public. A public library service in the North of England allows the public to look through its database to see if a particular book is stocked, and if so, which branch library to find it in. All these systems depend on *user-friendliness*. In viewdata systems, the screen display makes of lot of use of colour and graphics, so the Amstrad isn't an ideal terminal for these as the screen is monochrome. However if the service that you need to access operates under viewdata standards, then you *can* reach it, with the appropriate modem and software.

Accessing Viewdata Services

Viewdata services cannot be accessed using MAIL232. This is partly because viewdata uses characters outside the normal ASCII range, and partly because the information is sent in whole screenfuls, rather than character by character. Access is usually at 1200/75 baud (Prestel has a 300/300 *pstn* link in London), because more data is received than sent. Details of a viewdata communications software package are given in Chapter 9.

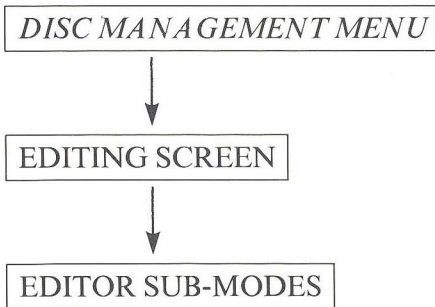
If the service that you want is operated over Prestel, access is usually by a local call over *pstn*. Some other services maintain private **nodes**, these are modems situated in various locations around the country which receive calls over *pstn* and then carry the data over a private network to the central computer. In either case, the user should have lower *pstn* costs.

Appendix 1

Menus

Have you seen a LocoScript menu and can't remember where it was? Here are the routes to the menus.

LocoScript has screens at different levels.



To go back to a higher level screen, press <EXIT>. To confirm a menu choice press <ENTER>. To get rid of an unwanted menu press <CANcel>.

DISC MANAGEMENT MENU

<C> = Create document

Create document
Name: DOCUMENT.000
Group: BOOK
Drive: A

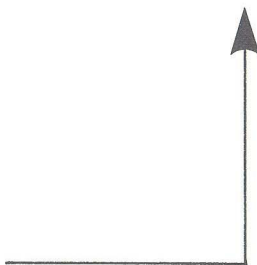
<P> = Print document

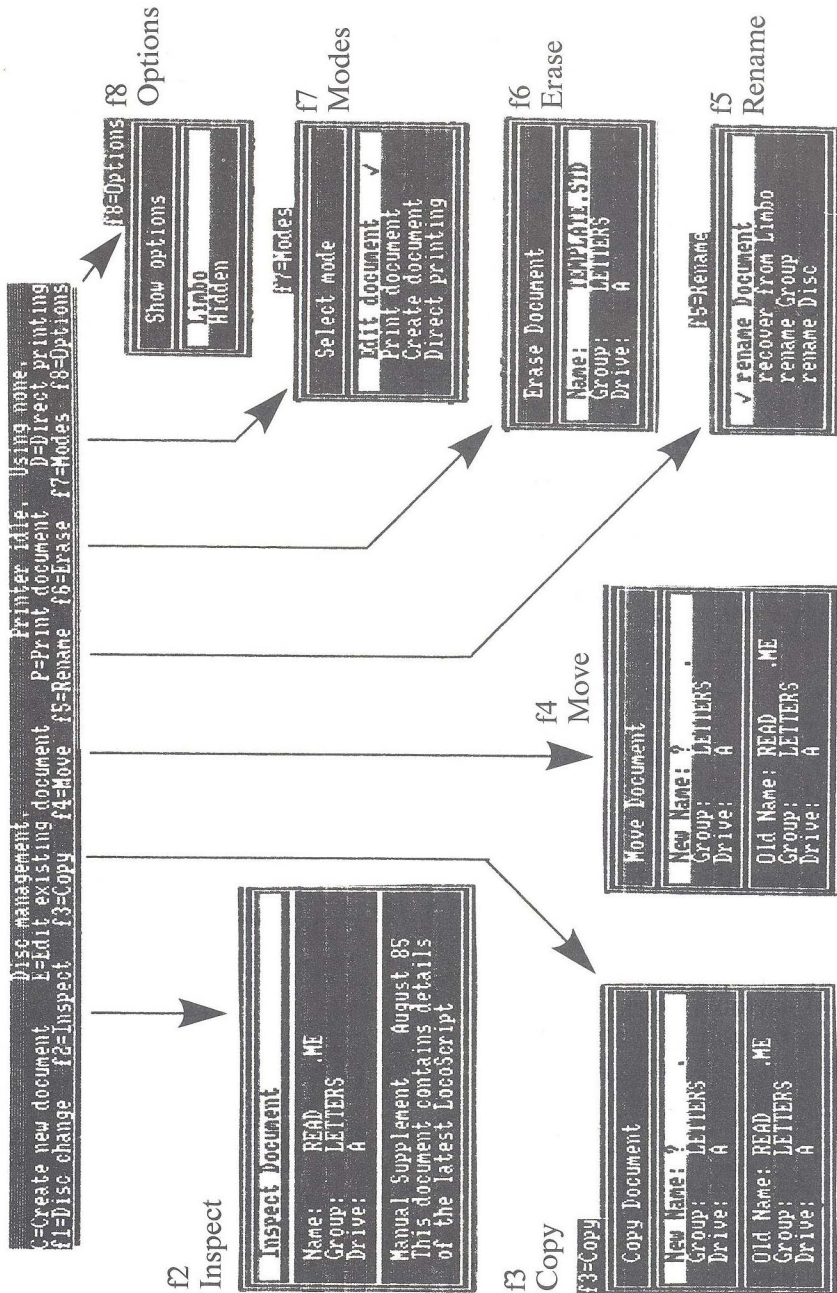
Print document
Name: READ .ME
Group: LETTERS
Drive: A
Print all pages
✓ Print some pages

<E> = Edit document

Edit document
Name: APPENDIX.1
Group: BOOK
Drive: A

Print some pages
Name: READ .ME
First page 1
From page 2
To page 2
Last page 3





EDITING SCREEN

f1=Show

Show state of:	
Codes	✓
Rulers	
Blanks	
Spaces	
Effectors	✓

f1
Show

f2=Layout

Document layout:	
Insert layout	
brand New layout	
Layout	??
Base layout	
Edit layout	
Layout	??

f2
Layout

f3=Emphasis

Emphasis codes:	
* Underline	
Full underline	✓
Word underline	
- Bold	
- Double	
- ReVerse Video	

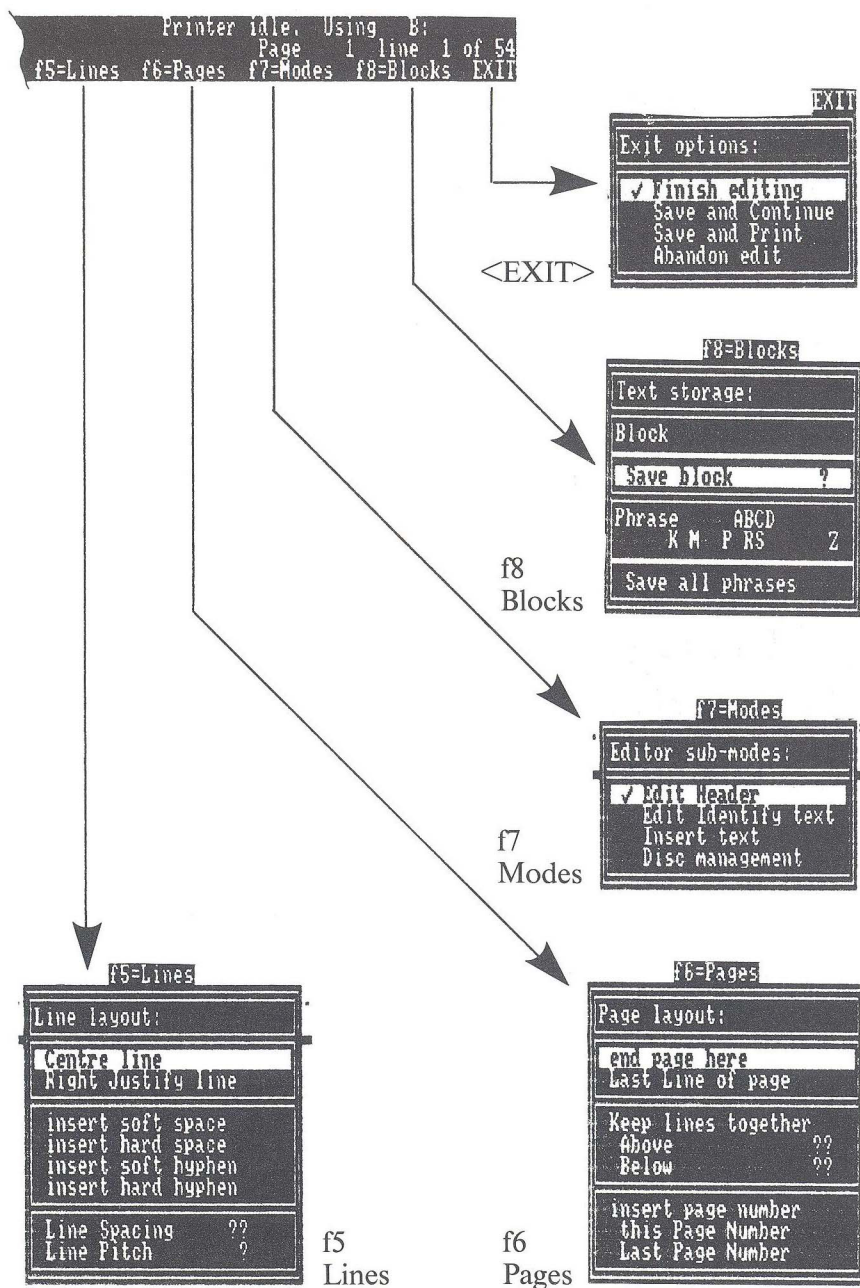
f3
Emphasis

B:\BOOK\APPENDIX.1 Editing text.
 -Layout -PiPS -LS2 -LP6
 f1=Show f2=Layout f3=Emphasis f4=Style

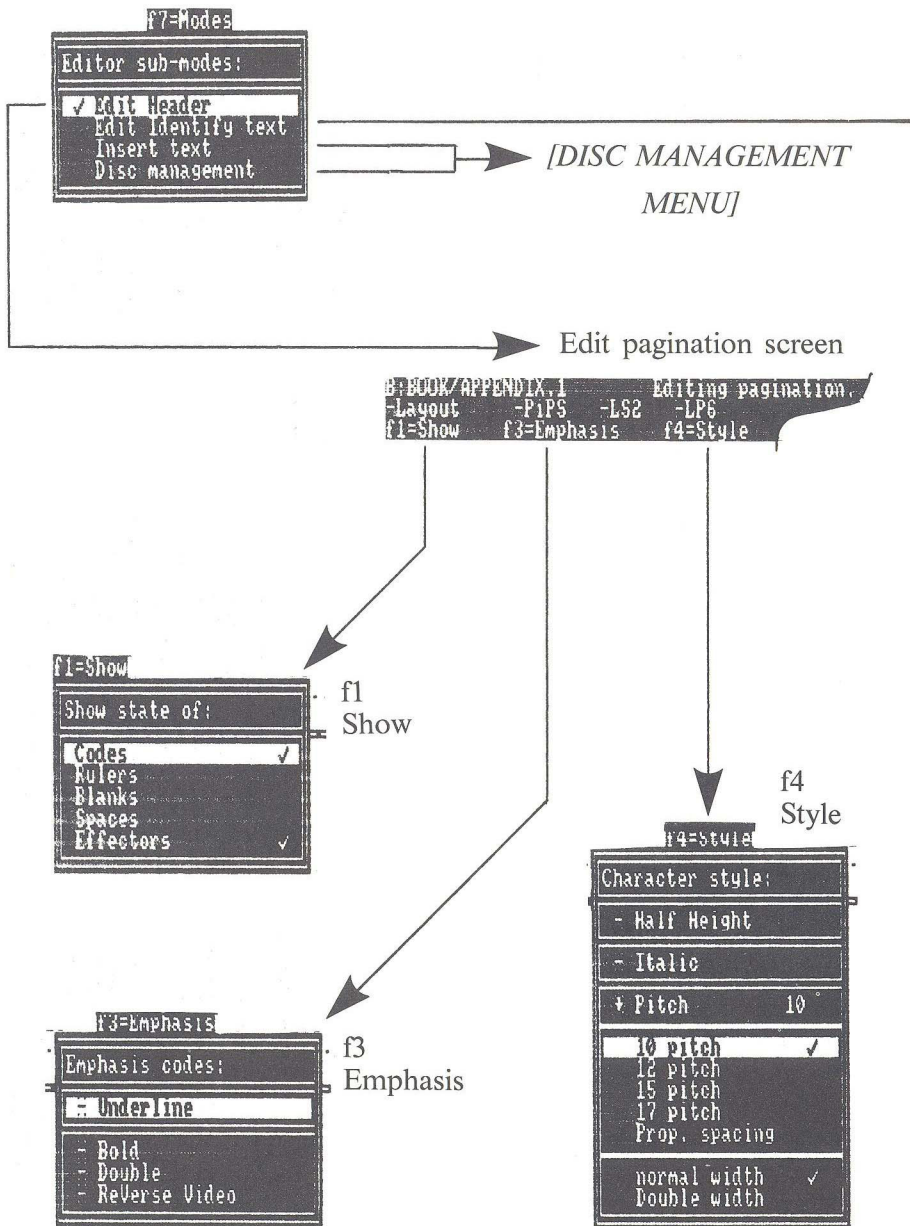
f4=Style

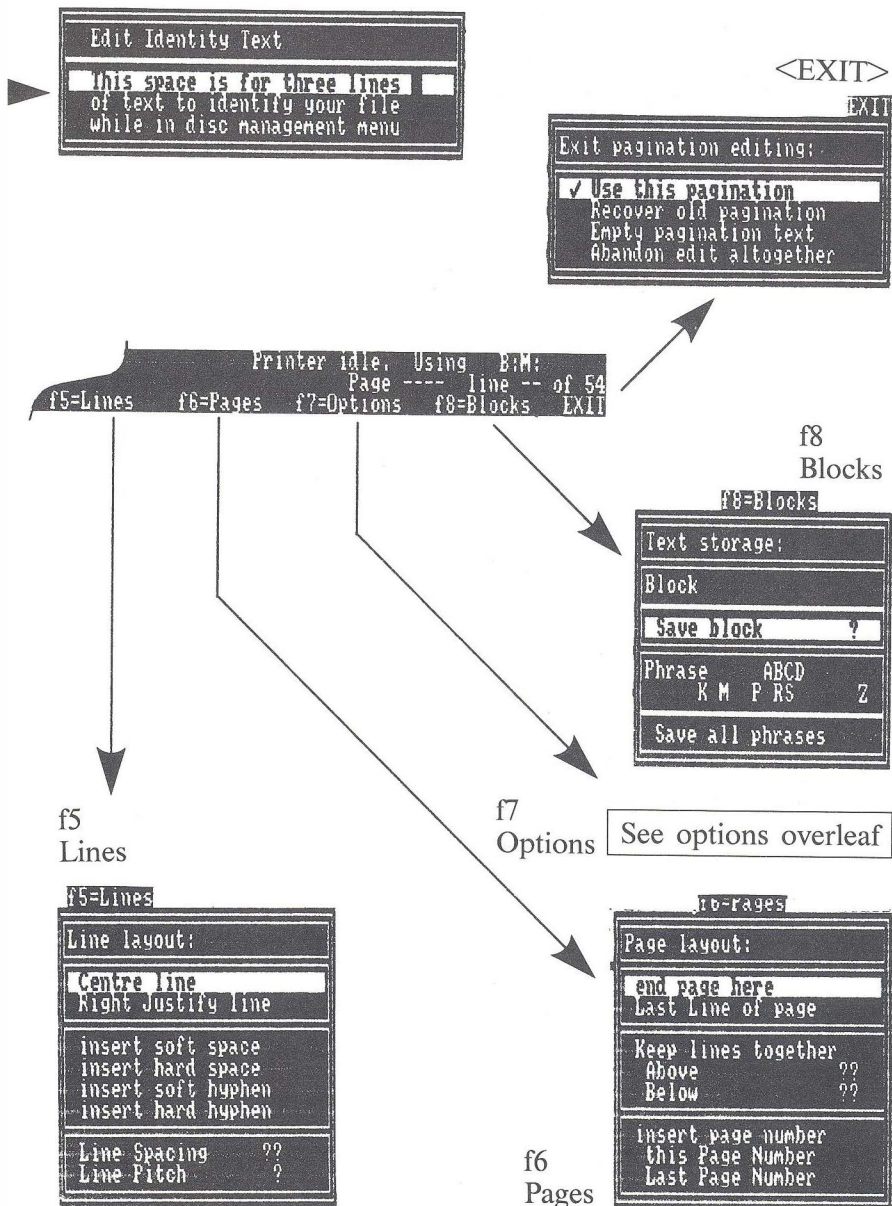
Character style:	
* Half Height	
SuperScript	✓
SubScript	
- Italic	
* Pitch	PS
10 pitch	
12 pitch	
15 pitch	
17 pitch	
Prop. spacing	✓
normal width	✓
Double width	

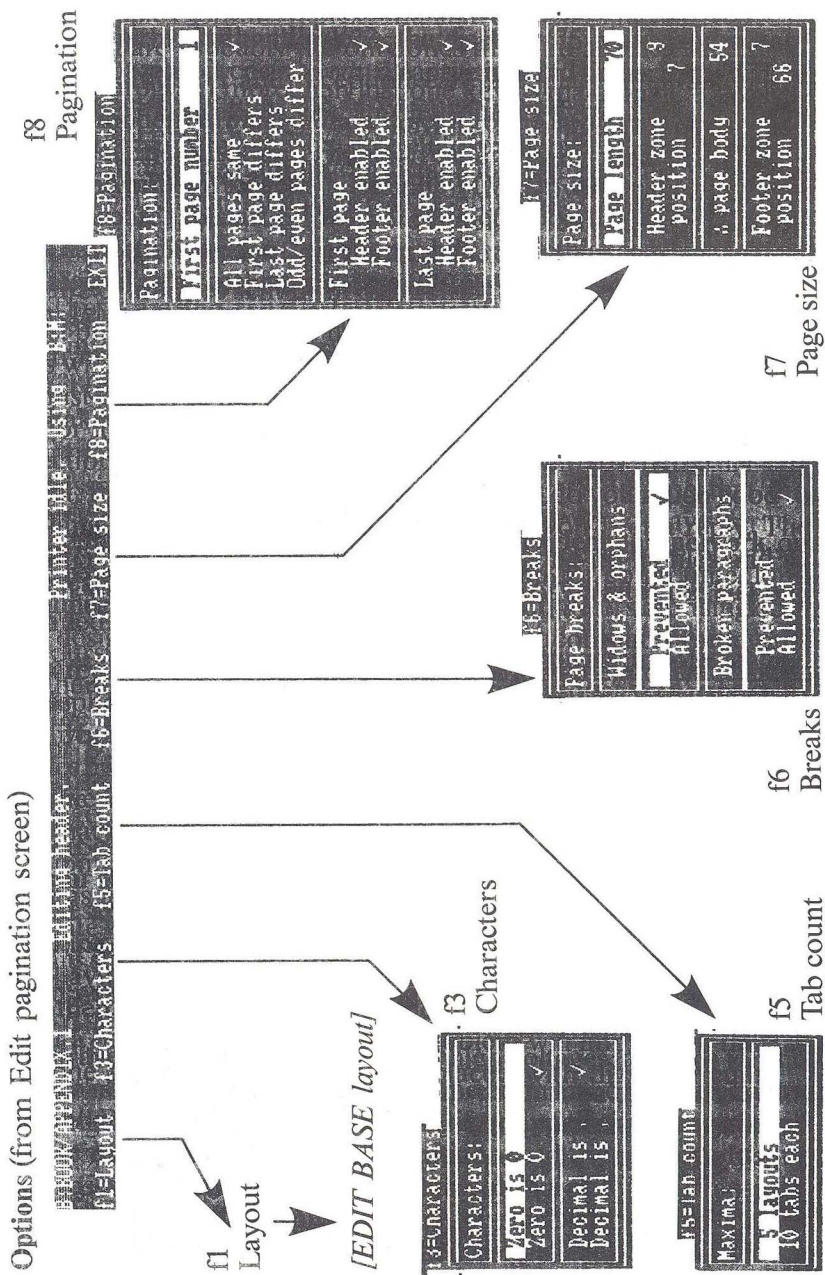
f4
Style



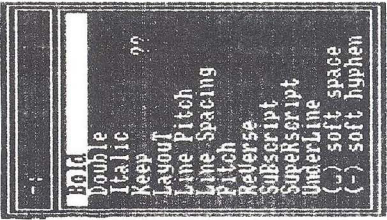
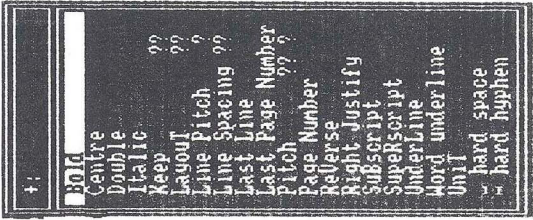
EDITOR SUB-MODES





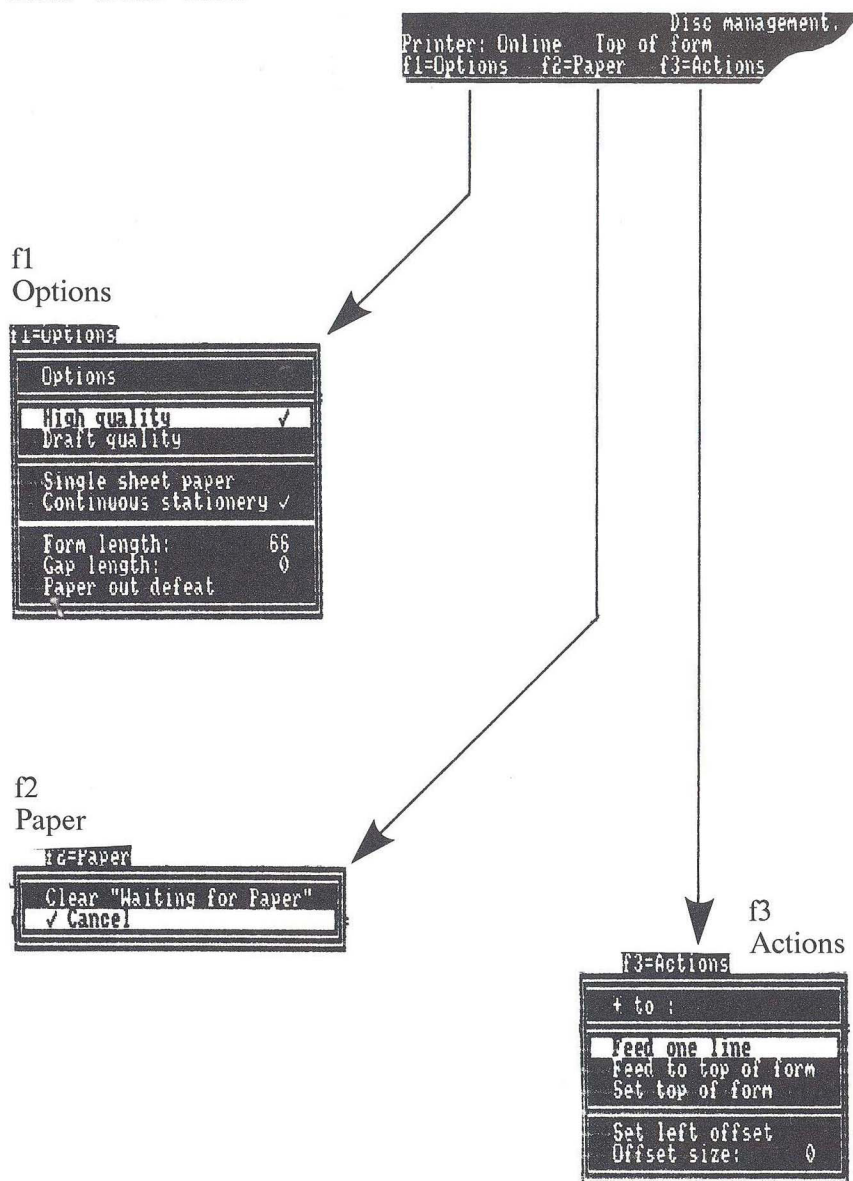


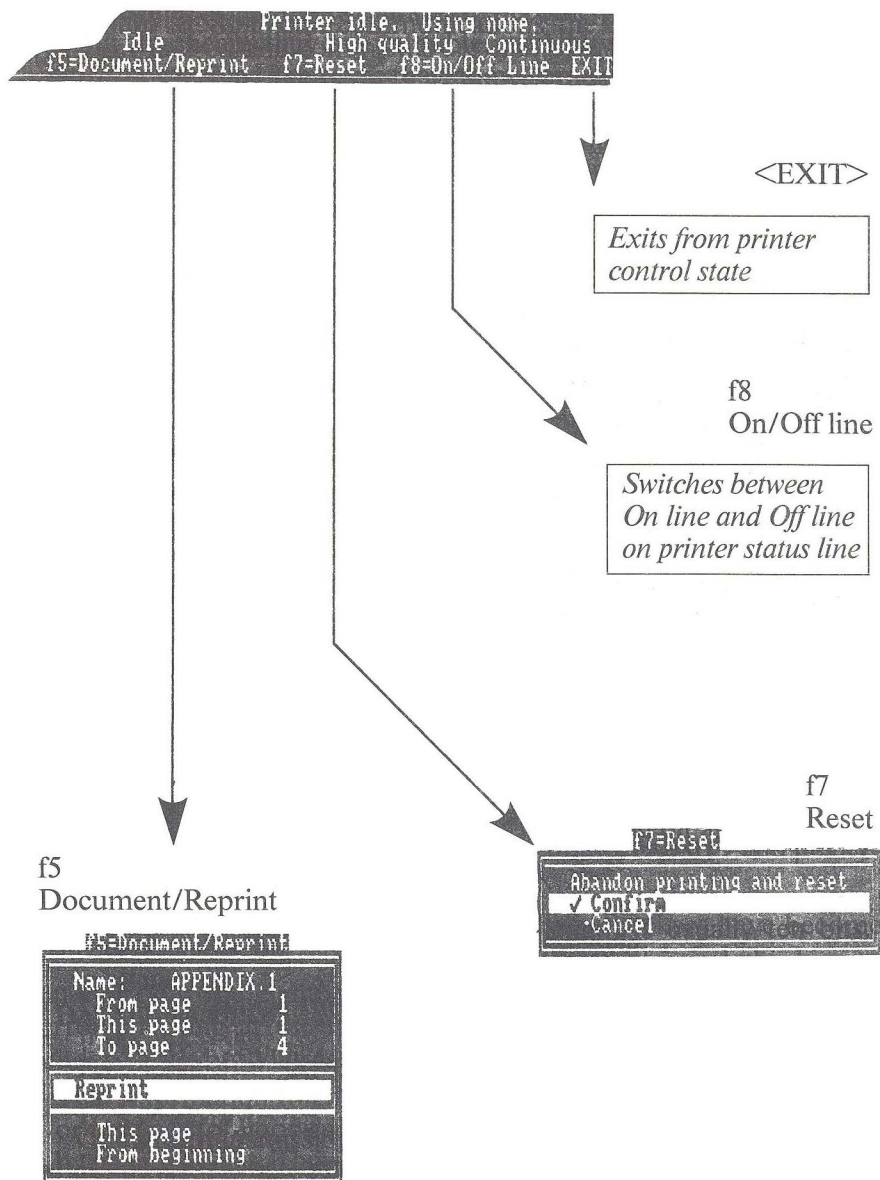
HELP MENUS (see Chapter 5)



PRINTER CONTROL MENUS (see Chapter 6)

Press <PTR> then:





Appendix 2

Character Sets

FULL CHARACTER SET UNDER LOCOSCRIPT

Number	Normal	Shift	Alt	Alt/Sft	Extra
1	1	!	½		ı
2	2	"	¼		--
3	3	£	⅜		℞
4	4	\$	½		¢
5	5	%	⅝		°
6	6	'	¾		˘
7	7	&	⅞		ˆ
8	8	*	Δ	Λ	˙
9	9	(æ	Æ	Θ
10	0)	ø	Ø	0
11	-	_	±		~
12	=	+	≈	≡	≠
13	q	Q	Θ		
14	w	W			
15	e	E	€		
16	r	R	ρ		⊗

Number	Normal	Shift	Alt	Alt/Sft	Extra
17	t	T	τ		™
18	y	Y	ψ		¥
19	u	U	↑		↑
20	i	I	⊗	⊙	
21	o	O	ω	Ω	⊖
22	p	P	π	Π	⏏
23	[{			
24]	}			
25	a	A	α		Ⓐ
26	s	S	σ	Σ	Ⓑ
27	d	D	δ	Δ	†
28	f	F	φ		f
29	g	G	γ	Γ	
30	h	H	←		
31	j	J	⊖		
32	k	K	→		
33	l	L	λ		
34	;	:		∴	↑
35	§	<	≤	≠	«
36	#	>	≥	⇒	»
37	z	Z			
38	x	X		χ	
39	c	C			©
40	v	V			
41	b	B	β		Ⓑ
42	n	N	↓		

Number	Normal	Shift	Alt	Alt/Sft	Extra
43	m	M	μ		
44	,	,	ç	Ç	
45	.	.	°	•	!
46	/	?	÷	×	¿
47	½	@	∞		\

THE ASCII CODE

Full Character Set for Data Communications

1	--	27	--	53	5	79	O	105	i
2	--	28	--	54	6	80	P	106	j
3	--	29	--	55	7	81	Q	107	k
4	--	30	--	56	8	82	R	108	l
5	--	31	--	57	9	83	S	109	m
6	--	32		58	:	84	T	110	n
7	--	33	!	59	;	85	U	111	o
8	--	34	"	60	<	86	V	112	p
9	--	35	_	61	=	87	W	113	q
10	--	36	\$	62	>	88	X	114	r
11	--	37	%	63	?	89	Y	115	s
12	--	38	&	64	@	90	Z	116	t
13	--	39	'	65	A	91	[117	u
14	--	40	(66	B	92	\	118	v
15	--	41)	67	C	93]	119	w
16	--	42	*	68	D	94	↑	120	x
17	--	43	+	69	E	95	`	121	y
18	--	44	,	70	F	96	#	122	z
19	--	45	-	71	G	97	a	123	{
20	--	46	.	72	H	98	b	124	!
21	--	47	/	73	I	99	c	125	}
22	--	48	0	74	J	100	d	126	~
23	--	49	1	75	K	101	e	127	0
24	--	50	2	76	L	102	f		
25	--	51	3	77	M	103	g		
26	--	52	4	78	N	104	h		

Appendix 3

Useful Addresses

Amsoft
Brentwood House
169 Kings Road
Brentwood
Essex CM14 4EF
Tel 0277 230222

Amstrad Consumer Electronics plc
Address as above
Tel 0277 228888

Amstrad User Software Database
PO Box 11
Gosforth
Newcastle upon Tyne
NE3 1RP

Data Protection Registrar
Springfield House
Water Lane
Wilmslow
Cheshire SK9 5AX
Tel 0625 535777

Digital Research (UK) Ltd
Unit 12, Fenton Way
Southfields
Basildon
Essex SS15 6SL

MicroLink
Database Publications
Europa House
68 Chester Road
Stockport
Cheshire SK7 5NY
Tel 061 456 8383

One-to-One
Scorpio House
102 Sydney Street
London SW3 6NL
Tel 01 351 2468

Prestel
Tel 100
(ask for *Freephone Prestel Sales*)

Sagesoft plc
NEI House
Regent Centre
Newcastle upon Tyne
NE3 3DS

Telecom Gold
60–68 St Thomas Street
London SE1 3QU
Tel 01 403 6777

Textnet
10 St Cross Street
London EC1N 8UB
Tel 01 242 2082

Index

abandon edit	76
abandon printing	86
abbreviations	71
Amstrad word processor	15, 20
ASCII	147
ASCII files	52, 167
auto-answer	153
back-up copies	37, 77
base layout	57
BASIC	99
BASIC programs	100
baud rate	149, 163
BBC Micro	172
blocks	52, 72
bulletin boards	181
buying discs	35
cables	28, 153, 173, 176
CAN (cancel) key	55
CCITT	150
centre tab	63
Centronics	80
character set	99, 201
codes	70
commercial software	129
communications disc	160
continuous paper	82

COPY key	74
copying discs	40
CP/M	22, 80, 87, 89
CP/M files	90
CP/M utilities	88, 91
CPS8256	144, 171
creating documents	51
cursor control keys	55
CUT key	74
data	145
data bits	145, 148
data communications	143
data file	101
databases	131, 187
decimal tab	63
deleting files	45
destination disc	41
device.com	92
dir.com	92
direct printing	87
directory	92
disc management menu	49
disc name	46
disckit	38, 93
discs	33
DOC key	55
documents	49, 51, 55
draft quality print	79
editing documents	69
editing screen	49
effectors	70
electronic mail	183
electronic typewriter	179
ENTER key	55
EOL key	55
erase.com	93
error checking	148
EXCH (exchange)key	169

EXIT key	75
file discs	37
file header	58
files	18, 24
finish editing	75
floppy discs	33
footer	65
footer position	65
footer zone	65
form length	82
formatting discs	39
formatting while copying	41
function keys	54, 162
gap length	82
group	53
group names	46
group template	57
hardware	21, 144
hardware handshaking	150, 163
header, file	58
header, page	64
header position	65
header zone	65
hidden files	159
high quality print	79
information	17
input	17, 20
inserting text	75
IPSS	157
italic	60
justify	60
lap portable computer	175
layout	56
left margin	61

left offset	85
letter pitch	59
limbo files	45
line pitch	59
line space	59
loading paper	30
LocoScript	49
LocoScript files	51
mail 232	145, 159, 171
mailbox services	183
margins	61
memory	18, 21
memory disc	23, 77, 89
menus	189
MicroLink	184
modem	152
new layout	63
on-line services	181
optional drive	15, 24, 34, 40, 42
options	81
output	17, 21
page header	64
PAGE key	55
page length	64
page numbers	66
pagination	67
paper.com	88, 94
paper out defeat	83
parameters, printer	81
parameters, terminal	161, 163
parity	148
PASTE	74
phrases	72
PHRASES.STD	73
pip.com	41, 42, 94, 160
preparing discs	36

printer parameters	81
printer ribbon	29
printing out	85
profile.sub	98, 117, 165
program	18, 100
program file	18
programmable function keys	162
PSS	155
PTR	81
pull-down menus	56
RAM	23
random access memory	15, 23
read only file	95
read/write file	96
receiving a file	168
recovering files	45
rename	46, 47
right justify	66
right margin	61
right tab	63
rped	117, 118, 164
RS232	21, 80, 148, 180
ruler	61
save and continue	76
save and print	76
second drive (optional on 8256)	15, 24, 34, 40, 42
sending files	168
serial interface	144, 171
serial transmission	146
set.com	95, 117, 118, 167
setkey.com	96, 164
setting up	28
show	61, 71
single sheet paper	82
software	21, 26, 129
source disc	41
spreadsheets	133
standard template	56, 68

standards	147
start of day disc	43
status lines	49
stop bit	147
submit.com	97
system file	93, 95
tab	62
Telecom Gold	165, 184
TEMPLATE.STD	57, 68
temporary stop	86
terminal parameters	161, 163
text file	18
text services	187
top of form	84
transient program area	23
transmission speed	149
type.com	98
unformatting	169
V.21	151, 181
V.23	151, 182
verify	43
Viewdata	147, 188
waiting for paper	88
word processor	16, 130
work surface	32
write-protect holes	36, 41
X on/X off	150

Using the Amstrad Word Processor

Many people need to write – for a living, as a hobby, or as part of some other interest. They would use a word processor if only they were cheap enough and easy to use. Many small businesses would use computers if only they were cheap enough, if there were suitable programs available and if they were easy to use. Many people would use electronic mail to communicate instantly with distant contacts, if only...

The Amstrad Word Processor is available, it's cheap, and – with this book – it's easy to use for all those tasks. Specially written for newcomers to computing, *Using the Amstrad Word Processor* will tell you how to use your Amstrad as a word processor, as a computer and as a communications terminal. And if you're not sure exactly what these things are, it'll tell you that too!

So, if you need to use your Amstrad to write articles, lecture notes, sermons, recipes, reports, or even *billets doux* to the milkman, then this is the book for you. Michael Milan is a video producer who has used his Amstrad to write scripts, articles, letters and this book. His other successful books are *A Young Person's Guide to BBC BASIC* and *Disk-Drive Projects for Micros*.

“a commendably clearly written guide” –
Small Business Confidential

NCC THE
NATIONAL CENTRE
FOR INFORMATION
TECHNOLOGY